قواعد البيانات

DB2

دكتور

أحمد أبو الفتوح صالح

مدرس نظم المعلومات

كلية الحاسبات والمعلومات

جامعة المنصورة

٢٠٠١/٢٠٠٢

المقدمة

تعتبر نظم إدارة قواعد البيانات من التطبيقات المعاصرة التي رافقت وارتبطت بالتطورات الحديثة في إستخدامات الكمبيوتر . كما يعتبر تخطيط قاعدة البيانات للمنظمة من الركائز الجوهرية إذا أرادت المنظمة أن تحقق العوائد التى تسعى إليها من إدارة موارد البيانات بها . كما أن أى خطة لقاعدة البيانات يجب أن تعد لمساندة الخطو الشمولية لنظم المعلومات بالمنظمة التى بدورها تساند الأهداف العامة والشمولية للمنظمة ذاتها .

وتعد نظم قواعد البيانات من النظم التى تتطور بشكل سريع ومتزايد بما ترتكز عليه من أدوات ومفاهيم حديثة لتخزين وتشغيل البيانات وإمكانية التحاور معها من أجل تلبية إحتياجات المستخدم المتعددة بالشكل الذى يخلو من أى تكرارية فى البيانات مما يساعد على تكاملية ومركزية البيانات . أى أن قاعدة البيانات هى الحل لكثير من مشاكل المعالجة التى تواجهها مركز معالجة البيانات فى المنظمات المختلفة .

ويمكن القول بأن بزوغ نظم قواعد البيانات قد ساهم الى حد كبير فى التخلص من جميع المشاكل التى إتسمت بها الملفات والنظم التقليدية لمعالجة وتشغيل البيانات ، حيث ظهر وتطور فى الآونة الأخيرة العديد من تطبيقات نظم إدارة قواعد البيانات التى تساهم الى حد كبير من جميع المشاكل القائمة فى النظم التلقليدية . ولقد سبق أن تعرضنا فى الجزء الأول من هذا المنهج الى الى مفهوم أسلوب قاعدة البيانات ، وأهم أهدافها وخصائصها ، وكذا مفهوم ومكونات نظم إدارة قواعد البيانات وأهم اللغات المستخدمة فى بيئة قواعد البيانات من أجل الرد على أى إستفسارات من قبل مستخدمة قاعدة البيانات . وسوف نتعرض من خلال هذا الجزء تناول مفاهيم قواعد البيانات المعاصرة وبنياتها الهيكلية وخصائصها المتنوعة ، والنماذج الهرمية والمبنية على العلاقات والشبكية والموزعة التى تطورت إليها عملية بناء قواعد قواعد البيانات . وقد إستعرضنا بالتفصيل من خلال هذا الجزء لكل من قواعد البيانات المبينة على العلاقات

وقواعد البيانات الموزعة لما لهما من إنتشار كبير فى المنظمات المعـاصرة . وفيمـا يتعلق بتخطيط قواعد البيانات فقد عرضنا هنا لأهمية التخطيط لقواعد البيانات والبيئـة التنظيمية التى تتواجد فيها والمهام المختلفة فى تخطيط قواعد البيانات وعملية التطيـط من أعلى الى أسفل والتصميم التصاعدى من أسفل الى أعلى والمنهج الذى يتبـع فـى التخطيط مع ذكر حالة لتخطيط قاعدة بيانات إحدى المستشفيات . وأخيرا إنتهى الكـاتب بعرض أهم المفاهيم المتقدمة فى بيئة قواعد البيانات وبالأخص المفهوم الـذى يتعلـق بقواعد البيانات الشيئية ODB.

بناءا على ماتقدم يمكن عرض خطة الدراسة فى هذا المؤلف على النحو التالى :

الفصل الأول : المفاهيم الأساسية .

الفصل الثانى : النماذج والأبنية فى بيئة قواعد البيانات .

الفصل الثالث : قواعد البيانات العلائقية (الارتباطية)

الفصل الرابع : منهجية إستخدام التصميم فى قواعد البيانات العلاقية .

الفصل الخامس : التحكم فى عرض البيانات بإستخدام لغة الاستفسار SQL

الفصل السادس :قواعد البيانات الموزعة وشبكات الاتصال.

الفصل السابع : تخطيط قواعد البيانات

الفصل الثامن : مفاهيم متقدمة فى بيئة قواعد البيانات .

الفصل التاسع : تطبيقات متقدمة بإستخدم قاعدة البيانات Access

وأخيرا ، نرجو من الله عز وجل أن يوفقنا فى تغطية كل جزء بشـكل متكـامل وسهل وبطريقة علمية واضحة فى تسلسل منطقى مقبول وبما يفيد القارئ .

والله الموفق

المؤلـف ،،،،

الفصل الأول

المفاهيم الأساسية فى بيئة قواعد البيانات

الفصل الأول
المفاهيم الأساسية فى بيئة قواعد البيانات

مقدمة :

تعتبر قواعد البيانات المرتبطة بالحاسب الإلكترونى من أحدث الأساليب المعاصرة لتخزين وإسترجاع المعلومات فى تطبيقات المعالجة الإلكترونية للمعلومات ، وبصفة خاصة فى المجالات التجارية والصناعية حيث تتزايد أهمية إستخدامها فى السنوات المقبلة لمواجهة تنظيم الكميات الهائلة من أحجام البيانات المرتبطة بالمشروعات الكبرى والأخذ بأحدث أساليب الإدارة الحديثة ، وتساعد قواعد البيانات فى مجموعة من المزايا أهمها :

- تخزين جمع البيانات لكافة الأنشطة فى منشأة ما بطريقة متكاملة ودقيقة وتصنيف وترتيب هذه البيانات بحيث يمكن إسترجاعها فى المستقبل .
- متابعة التغيرات التى تحدث فى البيانات المخزنة وإدخال التعديلات اللازمة عليها حتى تكون دائما فى الصورة الملائمة لإستخدامها فور طلبها .
- يمكن لقواعد البيانات تخزين كم هائل من البيانات التى تتجاوز الإمكانيات البشرية تذكر تفصيلاتها ، ومن ثم إجراء بعض العمليات والمعالجات التى يستحيل تنفيذها يدويا .
- تساعد قواعد البيانات على تخزين البيانات بطريقة متكاملة . بمعنى الربط بين النوعيات المختلفة للبيانات المعبرة عن كافة الأنشطة .
- تساعد قواعد البيانات فى تحقيق السرية الكاملة للبيانات المخزنة بها بحيث لاتتاح أى معلومات لأى شخص إلا لمن له حق الإطلاع عليها .

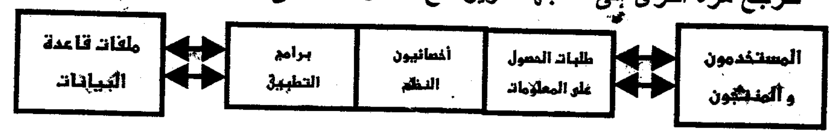
(١) مفهوم قواعد البيانات :

أصبح إصطلاح قاعدة البيانات Data Base شائع الإستخدام فى أوائل السبعينيات ، وفى السنوات التالية أصبحت قواعد البيانات واسعة الإنتشار وتزايدت أهميتها يوما بعد يوم . وسوف يكون بناء وتطوير قاعدة البيانات أحد الأنشطة الأكثر أهمية فى تصميم نظم المعلومات المرتبطة بالحاسب الإلكترونى فى السنوات القادمة . وليس لإصطلاح قاعدة البيانات تعريف قياسى دقيق . والتعريف العام هو :

قاعدة البيانات Data Base هى مخزن لكافة البيانات ذات الأهمية والقيمة بالنسبة للمستفيدين من نظام معالجة المعلومات Information Processing Systems .

وبعبارة أخرى يمكن أن يطلق على قاعدة البيانات بأنها " مجموعات بيانات موحدة تستخدم بواسطة نظم عديدة للمعلومات . ويعتمد شكل وسعة ودرجة تكامل البيانات على الإحتياجات النابعة من إدخال البيانات مضافا إليها إعتبارات التكلفة التى تتضمنها وتشتمل على تكاليف الحصول على البيانات وحفظها وتداولها ، والأبعاد الفنية المختلفة المتصلة بالوقت والسعة والعلاقات والصحة والأمن " .

أى أن نفس الأفراد المستفيدين من منتجات نظم المعلومات يعتبروا مسئولين أيضا عن إمداد قاعدة البيانات بالمدخلات . وتمر الطلبات المختلفة الخاصة بالمعلومات النابعة منها إلى أخصائيى النظم الذين يعدون برامج التطبيقات التى تتداول وتعالج البيانات عن طريقها ، أما المعلومات النابعة فترجع مرة أخرى إلى منتجها . ويوضح الشكل التالى هذا المفهوم :



شكل رقم (١) يوضح مفهوم تداول البيانات فى قاعدة البيانات

(٢) لماذا يتم الإستعانة بقواعد البيانات؟

تعد قاعدة البيانات ببساطة بمثابة أداة ذات غرض عام تسمح لك بتخزين أية معلومات معينة حتى يتسنى لك قراءتها وتحديثها لا حقا من خلال واحد أو أكثر من البرامج التطبيقية المتزامنة بطريقة آمنة ومعتمدة وعلى الرغم أن هذا التعريف يصف نظم قواعد البيانات التقليدية مثل SQL Server و Oracle 8i، إلا أنه يمكن الإستعانة به لوصف مجموعة من ملفات أقراص عادية .

٢-١ أساليب تخزين ومعالجة البيانات:

وهناك أساليب متعددة لتخزين المعلومات على الكمبيوتر ، إلا أن جميعها ينصب فى النهاية إلى وضعها فى أحد الملفات فى مكان ما على محرك الأقراص . وهناك نوعان أساسيان من الملفات ، واللذان تم تمييزهما وفقا لكيفية الوصول إليهما . النوع الأول يتمثل فى الملفات التى يتم الوصول إليها بصورة تسلسلية جملة واحدة . أما النوع الآخر فيتمثل فى الملفات التى يتم الوصول إليها بصورة عشوائية ، وهى تلك التى تتيح لك إمكانية الوصول إلى الملف مباشرة بصورة مختصرة .

فعند التعامل مع الملفات التى يتم الوصول إليها بصورة تسلسلية ، نكون بحاجة لتحميل الملف بأكمله داخل الذاكرة . ومن ثم ، تقوم العديد من البرامج بتحميل البيانات التى تكفى لأداء المهمة المعنية من الملف فحسب ثم تقوم بعدئذ بتحميل المزيد من البيانات عند إنتهاء المهمة . أى من قراءة البرنامج للبيانات من بداية الملف وحتى نهايته بصورة خطية ومن ثم جاء مصطلح تسلسلى بصورة.

ومن ناحية أخرى ، قد نرغب فى معظم البرامج التطبيقية فى الوصول إلى مجموعة فرعية من البيانات فى فترة معينة من الزمن ، لذا فبدلا من معالجة الملف بأكملة للوصول إلى جزء صغير من البيانات فحسب ، سيسمح لك نظام الوصول العشوائى إلى الملفات بتقسيم الملف إلى مجموعات يطلق عليها السجلات .

وسيوجد لدى كل سجل رقم سجل مميز ، والذى يتم تحديده من خلال موقعه النسبى فى الملف . ومن ثم ، يمكنك إسترجاع أو تحديث أى سجل فى الملف من خلال تحديد رقم السجل المتوافق .

تتمثل الفكرة الرئيسية وراء عمل نظام الوصول العشوائى إلى الملفات فى تشابة حجم جميع السجلات الموجودة فى الملف على نحو دائم ويسمح لك ذلك بتحديد موقع السجل من خلال حساب البايت الأول من السجل ، ويتم ذلك من خلال معرفة حجم السجل ورقمه النسبى فى الملف فحسب .

رغم أن الملف الذى يتم الوصول إليه بشكل عشوائى قد يتم تنظيمه بصورة مختلفة عن الملف الذى يتم الوصول إليه بشكل تسلسلى ، إلا أنه يمكن معالجة الملف الذى يتم الوصول إليه بصورة عشوائية كذلك بشكل تسلسلى من خلال قراءة السجلات وفقاً لترتيب أرقام السجلات الخاصة بها . يعنى ذلك أنك لن تفقد أية وظيفة من نظام الوصول للملفات بشكل تسلسلى فى حالة ما إذا إخترت أن تقوم بإنشاء الملف بحيث يتم الوصول إليه بصورة عشوائية .

٢-٢ تنظيم البيانات فى السجلات :

يمثل السجل مجموعة من البيانات الموجودة فى أحد الملفات ويتم تنظيمه داخل واحد أو أكثر من الحقول . إذ يوجد لدى كل حقل إسم معين ونوع بيانات مرتبط به . ويستخدم الإسم لتعريف الحقل فى السجل بصورة مميزة ، أما نوع البيانات فيصف نوع المعلومات التى يحملها الحقل .

ونظر لأن كل سجل فى الملف يحتفظ بنفس المعلومات على نحو دائم فإنه يمكن النظر إلى الملف الملئ بالسجلات بمثابة جدول أو شبكة ، حيث يتوافق كل سجل مع صف وكل حقل مع عمود وذلك كما هو موضح فى شكل (٢) . كما يشير مفهوم الجدول كذلك إلى مفهوم كيفية تخزين المعلومات فى الملف بصورة فعلية نظرا لأن الموقع النسبى لكل سجل

يضحى ثابتا بالنسبة للحقــول الأخــرى ، كمـــا أن رقــم الســجل النســبى يقــوم بتعريف موقع السجل فى الجدول بصـــورة ممــيزة .

Name	Address	Ctiy	State	Zip Code
Christopher James	1234 Main Street	College Park	MD	20742
Samantha Ashley	2345 Central Avenue	College Park	MD	20742
Terry Katz	3456 Pennsylvania Ave	College Park	MD	20742
Robert Weiler	4567 Redwood Way	College Park	MD	20742
Wayne Freeze	5678 Baltimore Street	College Park	MD	20742
Bonnie Jean	6789 Oak Street	College Park	MD	20742
Jill Heyer	7890 Washington Drive	College Park	MD	20742
Raymond Buky	8901 Souix Cirle	College Park	MD	20742

الشكل (٢) تشبه السجلات الموجودة فى الملف الشبكة أو الجدول

٢-٣ عمليـة الـتزامن :

تتمثل أحد أهم الأمور التى يجب وضعها فى الإعتبـــار عنــد التعــامل مــع أحد التطبيقات فى ضمان عدم تداخل إثنيـــن أو أكــثر مــن الــبرامج التــى يتــم تشغيلها فى آن واحد مع بعضها البعض أثناء معالجة البيانـــات . فـــإذا لــم يتــم إتخاذ الإجراءات الإحتياطيـــة ، ســيكون مــن الســهل علــى برنــامج أن يدمــر المعلومات التى قام برنامج آخر بتغييرها فـــى الملــف . علــى ســبيل المثــال ، إفترض أنه يوجد لديك ملــف يتــم الوصــول إليــه بصــورة عشــوائية يحتــوى على مراجعة أرصدة الحسابات الخاصـــة بـــالعملاء . ثــم إفــترض بعدئــذ إنــه يوجد لديك برنامج يؤدى إلى سحب الحســـابات الخاصـــة بالشــيكات التــى قــام العميل بتحريرها ، بينما يوجد برنامج آخر يعنـــى بإضافــة الأمــوال التــى قــام العميل بإيداعها . رغم أن السيناريو مختلف ، إلا أنـــه مــن الممكــن أن يحــاول البرنامجان تحديث نفس السجل للعميل فـــى آن واحــد .

تصـــور الآن التتــابع التــالى للأحــداث . ســيقوم البرنــــامج الخــاص بالمديونية كى يقوم بإجراء عملية تحديث المعلومـــات يقــوم البرنــامج الخــاص

بالإئتمان بقــــراءة ســجل العميــل وإضافــة الوديعــة إلــى الرصيــد الإجمــالى للحساب . عقب تحديــث المعلومـات الخاصــة بــالعميل مــن خــلال البرنــامج الخــاص بالإئتمــان ، بعدئــذ يقــوم البرنــامج الخــاص بالمديونيــة بإســتكمال العملية الخاصة به من خلال كتابــة رصيــد الحســاب الخــاص بــه والــذى تــم تحديثه مسبقا فى الملف . فى هـــذا الســيناريو ، تــم فقــد الوديعــة التــى تــم إضافتها للحساب ، نظرا لأن التحديث الــذى تــم مــن خــلال البرنــامج الخــاص بالمديونية إستخدام رصيد حســاب متقــادم .

تتمثل أفضل الطرق كى تحــول دون وقــوع مشــكلة الــتزامن فــى التحكــم فى عملية الوصول للبيانات ، ويتم ذلــك بصــورة نموذجيــة مــن خــلال تــأمين الملف بحيــث يعــوق المســتخدمين مــن الوصــول إلــى أى بيانــات مطلــوب إستخدامها فى الملف ، وبذلك يقــع تتــابع الأحــداث فــى الســيناريو المذكــور أعلاه كما يلى : يقوم البرنامج الخاص بالمديونيــة بتــأمين الملــف . ثــم يقــوم بعدئذ بقراءة سجل العميــل . ســيحاول البرنــامج الخــاص بالإئتمــان أن يقــوم بتأمين الملف . بيد أنه نظـــرا لأن البرنــامج الخــاص بالمديونيــة قــام بــالفعل بتأمين الملف ، فســيتم إيقــاف البرنــامج الخــاص بالإئتمــان لفــترة مؤقتــة ، ويستكمل البرنامج الخاص بالمديونية عملية التحديــث الخاصــة بــه ثــم يقــوم بإلغاء تأمين الملف . يستأنف البرنامج الخــاص بالإئتمــان بعئــذ ويتــم تحويــل التأمين إليه . ويقوم البرنــامج الخــاص بالإئتمــان عقــب ذلــك بــأداء عمليــة التحديث الخاص به ثم يقوم بإلغاء التأمين ، ومــن ثــم نضمــن وجــود الرصيــد الصحيح فى الملــف .

٢-٤ كيفيــة تــأمين البيانــات :

فى معظم نظــم الكمبيوتــر ، يمكنــك القيــام بتعريــف المســتخدمين الذيــن توجد لديــهم إمكانيــة قــراءة أو تحديــث ملــف معيــن ومنــع الآخريــن مــن الوصول إليه من خلال الإستعانة بالتــأمين العــادى الخــاص بنظــام التشــغيل . بيد أنه غالبا ما يفضل أن تسمح لشخص ما بمشـــاهدة جــزء واحــد فقــط مــن السجل – علــى ســبيل المثــال ، عندمــا ترغــب فــى الإحتفــاظ بالملعومــات

الخاصة بمرتب أحد الموظفين فى نفس السجل الذى يحتوى على عنوان الموظف ورقم تليفونه .

ولمنع الأشخاص من مشاهدة المعلومات الخاصة بالمرتب ، فإننا نضطر لإنشاء ملف آخر يحتوى على المعلومات الخاصة بكل شخص يوجد لدية إمكانية الوصول إلى الملف والحقول المسموح بالوصول إليها . ثم يتعين على البرامج التطبيقية بعدئذ أن تستخدم هذه المعلومات كما ينبغى حتى تحول دون الدخول على البيانات الغير مسموح بالوصول إليها .

مما لاشك فيه أن هذه الطريقة هى الطريقية الوحيدة الموثوق بها تماماً كما هو الحال مع فريق العمل الذى يقوم بإجراء برمجة التطبيق . إذ أن خطأ واحد صغير بالبرنامج قد يسمح للمستخدم بقراءة و/أو تغيير القيمة التى لا ينبغى الوصول إليها .

٢-٥ إنشاء النسخ الإحتياطية وإستخدام سجلات المعاملات :

هناك عدة طرق مختلفة تحول دون فقد البيانات . تتمثل أكثر هذه الطرق ذيوعاً فى عمل نسخة إحتياطية من البيانات بصورة مستمرة وبالطبع فأنك لن تسمح لأى مستخدم بإمكانية تحديث البيانات أثناء القيام بذلك ، أو سيسفر ذلك عن وجود ملفات البيانات بصورة غير متناسقة نظرا لأن شخص ما قام بتغيير القيمة فى ملف واحد قبل أخذ النسخة الإحتياطية منه وعقب أخذ إحتياطية من منف آخر . وتتضمن الطرق الأخرى التى تهدف إلى منع فقد البيانات الإستعانة بمحركات أقراص كثيرة للغاية و/أو وحدات خدمة قواعد بيانات ، ومن ثم ستظل البيانات موجودة حتى إذا فقدت محرك الأقراص أو وحدة الخدمة . رغم ذلك ، فنحن نوصى بشدة أن يتم عمل نسخة إحتياطية من البرنامج حتى فى حالة إستخدام أجهزة كثيرة للغاية ، وبذلك تكون فى الجانب الآمن فى حالة ما إذا فقدت كل من النسخة الرئيسية والنسخ الأخرى .

من ناحية أخرى ، تقوم معظم التطبيقات بعمل نسخة إحتياطية من البيانات مرة واحدة فى اليوم ، الأمر الذى يؤدى بدورة إلى تقليل إمكانية فقد البيانات إلى بيانات يوم واحد بحد أقصى . بيد أن فقد معلومات حتى نصف يوم كذلك يمكن أن يمثل مشكلة كبيرة فى معظم الأحيان . على سبيل المثال ، فى حالة وجود برنامج تطبيقى لمعالجة الطلبيات التى تتم عن طريق البريد الإلكترونى ، غالبا ما يوجد لديك مستندات الترتيب الأصلى ، مما يمكنك من إستعادتها مرة ثانية . فى المقابل ، فى حالة وجود تطبيق لمعالجة الطلبيات التى تتم عبر الهاتف ، قد لاتتعرف على العملاء الذين قاموا بعمل طلبيات فى حالة ما إذا فقدت الملفات ، وحتى إذا عرفت ، هل ترغب حقا فى طلب جميع العملاء مرة ثانية حتى تعلم ما الأشياء المطلوبة من قبل هؤلاء العملاء ؟

يوجد حلان لهذه المشكلة . أولا ، يمكنك طباعة كل تغيير تقوم بعمله للملفات على الطابعة . بهذه الطريقة ، سيوجد لديك دائما سجل مطبوع للتغييرات التى تمت ، ومن ثم يمكنك إستعادة المعلومات عند الحاجة إلى ذلك .

غير أنه هناك طريقة أفضل تتمثل فى كتابة جميع التغييرات فى ملف معين يطلق عليه سجل المعاملات . يحتوى هذا الملف على جميع التغييرات التى تمت لهذا الملف فحسب . من ثم ، إذا كنت مضطرا لإسترجاع الملف من النسخة الإحتياطية التى تم عملها فى اليوم السابق ، يمكنك تشغيل برنامج من شأنه أن يقوم بقراءة القيم من سجل المعاملات وإعادة تطبيقها على الملف الرئيسى .

غير أن كلا الطريقتين غير موثوق بهما بصورة تامة حيث أن مبرمج التطبيق ينبغى أن يقوم بإرسال المعلومات إلى الطابعة أو كتابة التغييرات داخل سجل المعاملات على نحو بين . فإذا لم يتذكر المبرمج أن يضم المكالمات الصحيحة فى البرنامج الخاص بها ، فلن تكون لهذه المعلومات ثمة جدوى .

٢-٦ قاعدة البيانات والحل الأمثل :

رغم أنه يمكن إستخدام الملفات كى ما تحتفظ بالبيانات الخاصة بالمؤسسة ، إلا أن إستخدام هذه الطريقة يمكن أن يسفر عن وجود عدة مشكلات . بيد أن نظام قاعدة البيانات يقوم بحل هذه المشاكل فضلا عن مشاكل أخرى قد تغيب عن ذهنك .

١/٦/٢ الجداول والصفوف والأعمدة :

تذكر أن قاعدة البيانات تحتفظ بالمعلومات فى جداول تتوافق مع الملفات التى يتم الوصول اليها بشكل عشوائى . إذ يتوافق كل سجل فى الملف مع صف فى الجدول ، وكل حقل فى الملف مع عمود فى الجدول . وعلى العكس من السجل ، يمكنك إختيار الأعمدة التى ترغب فى إسترجاعها فى الجدول . ومن خلال إسترجاع الأعمدة المطلوبة فقط ، أنت تساعد فى عزل البرنامج عن التغييرات الموجودة فى الجدول . يعنى ذلك أنه يمكنك إضافة وحذف الأعمدة من الجدول ، وطالما لا يشير البرنامج إلى أى من هذه الأعمدة ، فسيستأنف البرنامج العمل دون أدنى تغيير .

إن مفهوم الرقم النسبى للسجل غير متاح لتحديد موقع الصفوف فى الجدول . من ثم ، فقد تم إستبداله بمفهوم أكثر كفاءة معروف بإسم Primary Key . ويعد Primary Key بمثابة مجموعة من واحد أو أكثر من الحقول من شأن القيم الخاصة بها أن تقوم بتعريف صف بصورة مميزة فى الجدول . فكما إستخدمت رقم موظف أو رقم جزء بمثابة الرقم النسبى للسجل فى ملف يتم الوصول إليه بصورة عشوائية ، يمكنك الآن إستخدام مجموعة إحتياطية من القيم لتحديد موقع الصف ، هذا الأمر الذى يتيح لك المزيد من المرونة عند تصميم أحد الجداول .

٢/٦/٢ التـــأمين :

حيــث يقــوم نظـــام قـــاعدة البيانـــات أوتوماتيكيـــا بتحديـــد كيفيـــة إستخدام عمليات التأمين لمنع إثنين أو أكثر مـــن الـــبرامج مـــن الوصـــول إلى نفس البيانات فى ذات الوقت . وعلى النقيـــض مـــن التطبيقـــات التـــى تعتمد على الملفات بصورة أساسية حيـــث يتـــم تـــأمين الملـــف بأكملـــة ، فإن نظام قاعدة البيانـــــات يتسـم بأنـــه متطـــور بصـــورة كافيـــة تســـمح للبرامج بتحديث البيانات فى نفس الجـــدول فـــى آن واحـــد طالمـــا كـــانت جميع البرامج تصل جميعها إلى صفـــوف مختلفـــة .

لذا ، يمكنك الحصول على دعم عـــالى فـــى الأداء بالنســـبة للتطبيـــق على إعتبار أن قاعدة البيانـــات تســـمح لـــبرامج متعـــددة بـــالوصول إلـــى نفس الجدول فى آن واحد . أما بالنســـبة للتطبيقـــات التـــى تعتمـــد علـــى الملفات بصـــورة أساســـية ، فقـــد كـــان البرنـــامج الوحـــد يضطـــر الـــى استكمال عملية القراءة والمعالجة والتحديث قبـــل بـــدء البرنـــامج التـــالى. ومن ثم يستغرق البرنامج وقتا أطـــول بكثـــير كلمـــا كـــان هنـــاك المزيـــد والمزيد من المستخدمين الذيـــن يحـــاولون الوصـــول إلـــى الملفـــات فـــى نفس الوقـــت . ونظـــرا لأن نظـــام قـــاعدة البيانـــات يســـمح لتسلســـلات عمليات القراءة والمعالجة والتحديـــث المتعـــددة وغـــير المتضاربـــة بـــأن تكون نشطة فى آن واحد ، فلم يعد وقـــت الإنتظـــار بالتـــالى يمثـــل ثمـــة مشـــكلة .

٣/٦/٢ الأمـــن :

مبدئيا ، يتم تصميم نظـــم قواعـــد البيانـــات علـــى أســـاس أن تكـــون مؤمنـــة . فمـــن خـــلال نظـــام قواعـــد البيانـــات ، يمكنـــك تحديـــــد أى المستخدمين القادرين على الوصـــــول إلـــى أى بيانـــات وبـــأى أســـلوب . ونظرا لأنه تم إنتقال ســـمة الأمـــن خـــارج التطبيـــق ، فقـــد أصبـــح مـــن اليسير التأكد من أن المستخدمين المعتمديـــن فقـــط هـــم القـــادرون علـــى الوصول إلى جزء من المعلومـــــات .

٢/٦/٤ كيفية إستعادة البيانات المفقودة :

على النقيض من مجموعة من الملفات التى يتم الوصول إليها بصورة عشوائية ، يوجد لدى نظام قاعدة البيانات آلية شاملة تضمن أنه ما أن يتم كتابة البيانات داخل قاعدة البيانات لن يتم فقدها . ويتأتى ذلك نتيجة لأن نظام قاعدة البيانات يحتوى على أدوات من شأنها أن تقوم بعمل نسخة إحتياطية من قواعد البيانات بطريقة تحول دون وقوع مشكلة أثناء تسجيل بعض التغييرات ، وذلك على عكس ما يحدث فى النظم الأخرى . ويعنى ذلك إنه يمكنك إستعادة قاعدة البيانات بحالة متناسقة على نحو دائم .

فضلا عن ذلك ، يحتوى نظام قاعدة البيانات أيضا على سجل معاملات متكامل يقوم أوتوماتيكا بتسجيل جميع التغييرات المستحدثة على قاعدة البيانات . وعندما تكون مضطرا لإستعادة قاعدة البيانات ، ستتولى عملية الإستعادة مهمة معالجة سجل المعاملات أوتوماتيكيا . ويتم عندئذ إستعادة جميع المعاملات المكتوبة داخل سجل الأداء ، مما يعنى أن البيانات ستظل صحيحة فى أى وقت .

(٣) مزايا إستخدام قواعد البيانات :

يتوفر لقواعد البيانات عديد من الخصائص التى تميزها وتصفها وتفرق بينها وبين برامج التطبيقات المتفرقة وبذلك تكون المزايا التى تتصف بها نماذج قواعد البيانات الهرمية والشبكية والمبنية على العلاقات وفيما يلى ملخص لهذه الخصائص :(٤)

١- إستقلالية البيانات : Data independence

تعتبر هذه الخاصية من الدوافع الرئيسية التى أدت إلى تطوير وتطبيق نظم إدارة قواعد البيانات فى المنظمات المختلفة . حيث أن هذا المفهوم

يكون جليا لمبرمجى التطبيقات الذين يعدون الـــبرامج مـــن وجهـــة نظـــر تتصـــف بالمخطط الفكـــرى Conceptual Schema فـــى تنظيـــم البيانـــات . وبذلـــك تحفـــظ البيانات فى نطاق قاعدة البيانات بصفـــة مســـتقلة عـــن برامـــج التطبيـــق التـــى لا تتأثر بواسطة أى تغييرات فى تنظيم البيانــــات ماديـــا . كمـــا أن ذلـــك لا يســـتبعد أى تأثير راجع Retrospective على الـــبرامج ممـــا يقلـــل إلـــى حـــد كبـــير مـــن صيانة الـــبرامج .

ويحفظ نظام إدارة قواعد البيانات كـــل روابـــط البيانـــات الداخليـــة Internal data links وإستراتيجيات الفـــرز والوصـــول والتكـــامل المـــادى للبيانـــات علـــى الأقراص الممغنطة . وتؤدى هذه الوظيفة عـــن طريـــق قـــاموس البيانـــات Data Dictionary . مـــا يحـــرر المـــبرمجين مـــن الإجـــراءات الروتينيـــة ويركـــزون عملهم على تحسين مـــدى منطقيـــة الأداء .

٢- مركـــزية البيانـــات : Data Centralization

توضح هذه الخاصية أن نموذج قاعدة البيانـــــات يحفـــظ شـــكل واحـــد مقنـــن للبيانات بدلا من أشكال عديدة . وقـــد طـــورت خاصيـــة المركزيـــة كـــإمتداد لنظـــم المعلومات حيث يحفظ كل تطبيـــق ملفاتـــه الخاصـــة فـــى إطـــار مركـــزى موحـــد وتعتبر الوظيفـــة الأساســـية لإدارى قـــاعدة البيانـــات Database Abministrator (DBA) تأمين سلامة وأمـــن البيانـــات ومتابعـــة مركزيتـــها الموحـــدة للحـــد مـــن التكرار والحشو الذى لا داعى لـــــه .

٣- تقليـــل الحشـــو : Reduction of redundency

تختص هذه الخاصية بالحد من التكـــــرار وتقليـــل الحشـــو والاســـهاب الـــذى كان سائدا من قبل فى برامج التطبيقـــات المتواجـــدة فـــى نظـــم معالجـــة البيانـــات وبإستخدام هذه الصفة أمكن الوصـــــول إلـــى :

- تجنب عـــدو التوافـــق Inconsistency .
- الحد من حفظ نسخ عديد من نفـــس البيانـــات .

- حذف البيانات الزائدة عـــن الحاجـــة .

- توفير مساحة التخزين بحذف الحشـــو والتكـــرار .

- المشاركة فى البيانات Data Sharing بين أكـــثر مـــن مســـتخدم .

٤- تكـامل البيانـات : Data Integrity :

تتميز نظم إدارة قواعد البيانـــات بخاصيـــة التكامليـــة لوحـــدات بياناتـــها . حيث أن الفعــــل أو التصـــرف Transaction تعتـــبر الوحـــدة النوويـــة Atomic Unit أصغر وحدة لتنفيذ تتابع مجموعة مـــن العمليـــات تـــؤدى فـــى شـــموليتها Entirety أو لا تنجـــز مطلقـــا . أى أن فعـــل أو تصـــرف مـــا يكمـــل فعـــــل أو تصرف آخر لكى يقود إلى معرفة وتحديد معالم عمليـــة محـــددة كمـــا فـــى حالـــة تصرفات الدائن والمدين التى تكـــون رصيـــد أحـــد عمـــلاء البنـــك علـــى ســـبيل المثال . يتضح من ذلك أن التصرف الأصغـــر يعتـــبر مـــن الوســـائل المســـتخدمة التى يمكن عن طريقها الحصـــول علـــى تكامليـــة قـــاعدة البيانـــات حيـــث أنـــها تؤكد أن كل التصرفات قد تحول قاعدة البيانـــــات مـــن حالـــة ثابتـــة إلـــى حالـــة أخرى منجزة أو أن الحالة الثابتة التمهيدية تركـــت ولـــم تمـــس . وبذلـــك فـــإن إصطلاح التكاملية يشير إلـــى :

(١) تنسيق البيانات بواسطة التطبيقـــات والتصرفـــات المختصـــة .

(٢) إمتداد تحديث البيانات لكى ينطبق علـــى البيانـــات الأخـــرى المتـــأثرة .

(٣) الإحتفاظ بدرجة من الإتســـاق والتصحيـــح .

٥- خصوصية وأمن البيانات : Data Privacy and Security :

فى إطار قواعد البيانات يمكن التأكد مـــن أن الوصـــول المعتمـــد هـــو الـــذى له الحق فـــى إســـتخدامها . فخصوصيـــة وأمـــن البيانـــات تعتـــبر مـــن الـــبرامج الروتينية الهامة التى يضمها نظام إدارة قواعد البيانـــات وبذلـــك يجـــب ألا تكـــون البيانات المختزنة فى قاعدة البيانات عرضـــة للفقـــد أو الســـرقة أو الإطـــلاع غـــير المعتمد أو غير المرخص بـــه . وتحـــدد قـــاعدة البيانـــات إجـــراءات الخصوصيـــة

من خلال البرامج الروتينية المتضمنة بها التى تشير إلى حقوق الأفراد والأنشطة فى داخل المنظمة من أن تحدد لنفسها متى ؟ وكيف ؟ ومن ؟ وإلى أى مدى ؟ يمكن أن ترسل البيانات للآخرين أو يكون لهم الحق فى الوصول إلى المعلومات المتضمنة بالقاعدة . كما تحدد إجراءات أمن البيانات وحمايتها ضد الإطلاع أو النشر ، المتعمد أو غير المتعمد من قبل الأفراد غير الرسميين والغير مصرح لهم أو ضد تزوير المعلومات أو سرقتها أو تلفها وما شابة ذلك .

٦- **جودة البيانات** Data Quality

يؤكد أسلوب نظام إدارة قواعد البيانات جودة ما يحوية من بيانات من خلال توفير عدة قدرات منها :

أ- **مجموعة إجراءات الصحة** : Set Validation

التى تسمح بتواجد معايير محددة تتصل بعلاقات البيانات معا فى إطار يتسم بالصحة والوثوق .

ب- **ترابط البيانات** : Relatability

يتضمن ذلك قابلية توفير العلاقات بين السجلات والمكونات بطريقة مناسبة . وتعتبر العلاقات مهمة وقابلة للتقنين مثل أى خاصية للبيانات والسجلات .

ج- **الإستدعاء** : Recovery

تتصل تسهيلات الإستدعاء بآلية إستدعاء المعامل أو المشغل الذى يختار عند توقف الجهاز أو ظهور خطأ فى برنامج التطبيق المستخدم وبذلك تتعامل هذه الخاصية مع مشاكل حفظ بيانات التصرف أو الفعل المعين .

د- **الضبط المتزامن** : Concurrency Control

يتعامل هذا المعامل مع التصرفات المتزامنة حيث أنه عن طريقه يمكن ملاحظة المشاكل بطريقة تزامنية Synchronization معا .

هـ- **تصحيح الأخطاء والتعديل** : Debugging and tuning

تقدم هـذه الخاصيـة ذات التفـاعل البينـى Interactive Interface القـدره لتصحيح البيانات الخاطئة فى قاعدة البيانـات بسـرعة كبـيرة . كمـا تسـاعد أيضـا فى تعديـل Tuning تطبيقـات قـاعدة البيانـات عـن طريـق إسـتخدام إختبـارات محددة مثل " ماذا إذا " << What If >> التى يمكن أن تـؤدى فـى نمـط تفـاعلى.

و- البسـاطة : Simplicity

تستخدم المؤشـرات Pointers فـى نظـم متعـددة لإظـهار العلاقـات بيـن مفردات البيانات مما يقدم وجهة النظـر المنطقيـة التـى تسـهل الـترتيب والعـرض والفـهم .

ز- توفير البيانـات : Data availability

بسبب تضخم حجم البيانات التـى تتضمنـها قـاعدة البيانـات فإنـها تحتـاج إلى طرق تداول سريعة تصبغ عليـها الأداء الجيـد والفعاليـة العاليـة .

٧- توفـير البيانـات : Data Availability

تقـدم نظـم إدارة قواعـد البيانـات مجموعـة مـن القـدارت العاليـة المتصلـة بتوفر البيانات وإمكانية البحث فيها . ومن هـذه القـدارات مـا يلـى :

أ- الوصـول المـتزامن Concurrent Access لقـاعدة البيانـات عـن طريق برامج مسـتخدمين متعددين Multi – Users .

ب- مسارات الوصـول المتعـدد للبيانـات Multiple Access التـى توفـر برامج تطبيق تؤدى إلى توفـير بيانـات أكـثر مـن طريقـة وصـول مفـرد لقاعدة البيانـات .

ج- أسـلوب غلـق البرنـامج نـهائيا Deadlock ويؤكـد ذلـك أن أسـلوب المعالجة المستمرة يحاول معالجة برنامج تطبيـق أو أكـثر لغلـق السـجلات التى لا يشارك فيها البيانـات المطلوبـة .

د- تصفح قاعدة البيانات DB scan تسـهل إمكانيـة التسـاؤل بـدون الحاجـة لكتابة برنامج تطبيق خاص بذلـك .

هـ- لغة التساؤل Query Language تساعد فى إنتاج الإجابة على التساؤلات وإخراج التقارير المطلوبة بدون الحاجة لتوفر خبرات مميزة متخصصة لأداء ذلك .

وبذلك تسهم خاصية توفير البيانات بقدراتها المتنوعة فى مساعدة الباحث على توجيه عدد من الأسئلة والتساؤلات ذات التنوع الواسع لما تحوية قاعدة البيانات من بيانات مختزنة فيها . وتكون هذه التساؤلات متدفقة ومصممة سلفا حتى يمكن التعامل معها بالبحث السريع . وبذلك تعتمد إمكانية البحث فى قاعدة البيانات على توفر معايير بحث مختلفة خاصة بالتنظيم الطبيعى للبيانات . ومع تواجد عدد كبير من تنظيمات البيانات فى إطار قاعدة البيانات فإن زمن البحث يكون أطول جدا لكى يستجيب للوقت الحقيقى المتاح عند النهايات الطرفية .

(٤) محددات إستخدام منهج قواعد البيانات :

ينطوى إستخدام منهج قواعد البيانات على بعض القيود تجعل من الأنسب فى بعض الحالات تطبيق منهج الملفات :

١- يرتبط تصميم قواعد البيانات بعاملى التكلفة والسرعة ، فعندما تختار المنظمة مصمم قاعدة بيانات متكاملة لها يتطلب ذلك دراسة مستفيضة وشاملة لاحتياجاتها الحالية والمستقبلية من البيانات ، وعلاقاتها ببعضها البعض ويستغرق ذلك وقتا وجهدا وتكلفة . وبالمقارنة بمنهج الملفات نجد أن الإحتياج للبيانات يدرس لكل تطبيق على حدة ، فتكون الدراسة أبسط ، كما تتنوع التكلفة بمرور الوقت كلما ظهرت الحاجة لتطبيق جديد .

٢- يجب أخذ تكاليف المكونات المادية والبرمجيات الخاصة بقاعدة البيانات فى الإعتبار وهى للحاسبات الكبيرة المركزية قد تصل لمبالغ ضخمة . ومن الشائع إستخدام حزم قياسية مثل IDMSأو ORACLE ، كما أنه يجب شراء كميات ضخمة من وسائط التخزين منذ البداية .

٣- الدخول لقاعدة البيانات أبطأ من الدخول المباشر للملفات ، حيث أن الدخول يكون من خلال طبقة أخرى من الرمجيات بين البرنامج التطبيقى والقاعدة ، ألا وهى نظام إدارة القاعدة وبإنخفاض تقنية الأقراص بمرور الوقت ، والتقدم فى سرعتها ، تتضائل هذه العيوب كثيرا ، وعموما يكون نظام الملفات المستقلة ملائما للمنشآت فى الحالات التالية :

- إذا كان التطبيق يتكرر بإستمرار مع كميات ضخمة من البيانات .
- إذا كانت بيانات المنشأة ليس من المتوقع أن تتغير على مدى طويل .
- إذا لم تكن متطلبات المرونة عالية ، بأن كانت المعلومات تنتج وفق معايير قياسية .

ولهذه الأسباب فإن منهج الملفات المستقلة شائع فى أعمال المحاسبة المالية وإمساك الدفاتر ، حيث يواجه معالجة مكثفة للمعاملات . أما الأعمال المرنة كالمحاسبة الإدارية ونظم دعم القرار فلا تناسب إلا أنظمة قواعد البيانات .

(٥) المتعاملون مع قواعد البيانات :

كل المتعاملون مع قواعد البيانات ثلاث فئات ، لكل منهم مستوى معين :

١- المشرف على قاعدة البيانات :

يشغل المشرف على القاعدة Database Administrator (DBA) مستوى وظيفيا عاليا فى أنظمة ويتطلب القيام بهذه الوظيفية الإلمام بمختلف فئات المستخدمين وكذلك معرفة تقنية شاملة . فمشرف القاعدة مسئول عن إدارة مصادر البيانات بأسلوب سلس وفعال ، مما يجعل مسئوليته ضخمة بالفعل . ويتضمن عمله عادة المسئوليات التالية :

- المساعدة فى أعمال التحليل والتصميم للقاعدة للوصول إلى المستوى التقنى المطلوب والحفاظ عليه .

- تحقيق المستوى المطلوب من الأمان للقاعدة ، **ويتطلب ذلك:**
  - ضمان تطبيق أكواد الترخيص للدخول لبيانات القاعدة .
  - ضمان وجود النسخ الإحتياطية الكافية .
  - وضع كافة القيود التى تضمن سلامة القاعدة .
- ضمان الإستخدام الأمثل للبيانات مع مراعاة التكلفة .
- إعادة التنظيم المادى للقاعدة كلما دعت الحاجة لذلك .
- تنميط طرق عرض البيانات والوثائق المتعلقة بها .
- التواصل مع مستخدمى القاعدة لتحقيق منطلباتهم .
- تدريب المستخدمين .

٢- مبرمجو التطبيقات :

وهم المسئولون عن وضع البرامج التطبيقية التى تستغل القاعدة وصيانتها . وتحتوى البرامج على الأوامر الأساسية التى تمكن من التخاطب مع القاعدة وذلك عبر نظام إدراتها ، وتحتوى " لغة التحكم في البيانات Data Manipulation Language DML" " على أوامر للتعامل مع البيانات مثل : التخزين Store والاسترجاع Retrieve والتعديل Modify والحذف Delete والإضافة INSERT . ويجب علي المبرمج الإلمام بقدر كاف بهيكل القاعدة ليتمكن من استخدام الأوامر المطلوبة بكفاءة .

٣- المستخدمون :

وهم الموظفون القياديون أو المكتبيون الذين يتعاملون مع القاعدة من خلال البرامج التطبيقية المتاحة لهم . وسابقا كانت التقارير تخرج في صور قياسية لا يد للمستخدم فيها ، ولكن أصبح للمستخدم بفضل برمجيات إدارة القاعدة حرية في الحصول علي ما يريدون باستجابة سريعة وفورية ، مما زاد من فعالية صنع القرارات .

(٦) البينـة الخاصـة بقواعـد البيانـات

يتم تنفيذ جميع نظم قواعد البيانـات المتاحـة فـي هـذه الآونـة تقريبـا مـن خلال بنية الوحدة التابعة / وحدة الخدمـة . وتقـوم هـذه البنيـة بتعريـف نوعيـن من البرامج وكيفية تفاعلهما الواحد مـع الآخـر .

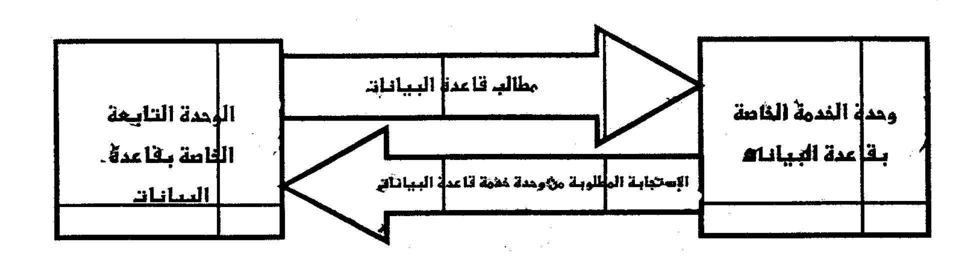
٦-١ وحـدات الخدمـة والوحـدات التابعـة

تعد الوحدة التابعة بمثابة برنـامج يقـوم بإنشـاء متطلبـات يتـم إرسـالها إلي برنامج آخر – يطلق عليـه وحـدة الخدمـة – بغـرض تنفيذهـا . وحينمـا تنتهي وحدة الخدمة من تنفيـذ المطـالب المعنيـة بـها ، تظـهر النتـائج فـي الوحدة التابعة . وفي البيئة الخاصة بأجـهزة الكمبيوتـر المتاحـة حاليـا ، يتـم تنفيذ العديد من التطبيقات من خلال تقنيـة الوحـدة التابعـة / وحـدة الخدمـة . علي سبيل المثال ، يمثـل متصفـح الويـب الوحـدة التابعـة المسـؤولة عـن التعامل مع وحدة خدمة الويب . وتحتوي وحـدة خدمـة الملـف علـي الملفـات المتاحة لك عبر الشـبكة . بطريقـة مماثلـة ، تسـمح وحـدة خدمـة الطابعـة لمدير الشبكة بمشاركة طابعة واحدة مع العديـد مـن المسـتخدمين .

ملحوظة : في الغـالب الأعـم ، يسـتخدم مصطلحـي الوحـدة التابعـة ووحـدة الخدمـة لوصف كل من تطبيقات البرامج والأجـهزة التـي يتـم تشـغيل الـبرامج مـن خلالها. وقد يؤدي ذلك إلي حـدوث بعـض اللبـس . فمـن الممكـن أنـك قـد تستخدم تطبيق متصفح الويب (الوحدة التابعة) للوصـول إلـي وحـدة خدمـة الويب علي نفس الكمبيوتر . علي سـبيل المثـال ، غالبـا مـا أقـوم باختبـار صفحات الويب الجديدة علي وحدة خدمة الاختبـار التـي يتـم تشـغيلها علـي نفس الجهاز مثل المتصفح ، وذلك قبل تحميلـها علـي وحـدة خدمـة الإنتـاج . وبوجه عام ، عندما اسـتخدم مصطلحـي الوحـدة التابعـة ووحـدة الخدمـة فأنا أشير بذلك إلي التطبيقات البرمجية لا إلـي المكونـات الفعليـة التـي يتـم تشغيلها عليـها .

٦-٢ وحدة الخدمة الخاصة بقاعدة البيانات والوحدات التابعة الخاصة بقاعدة البيانات

تعتبر وحدة الخدمة الخاصة بقاعدة البيانات بمثابة برنامج يستقبل مطالب قاعدة البيانات من خلال الوحدة التابعة الخاصة بقاعدة البيانات ، ثم يقوم بدوره بمعالجة هذه المتطلبات علي الوحدة التابعة الخاصة بقاعدة البيانات ، وذلك علي النحو الموضح في شكل (٣) . علي الجانب الآخر ، تعتبر الوحدة التابعة الخاصة بقاعدة البيانات بمثابة برنامج يقوم بإنتاج مطالب قواعد البيانات من خلال التفاعل مع المستخدم أو معالجة ملف البيانات أو الاستجابة إلي حدث معين في الكمبيوتر . ويمثل مطلب قاعدة البيانات عملية معينة يتم القيام بها من خلال وحدة الخدمة الخاصة بقاعدة البيانات ، مثل إظهار النتائج من خلال جدول أو تحديث واحد أو أكثر من الصفوف في جدول أو القيام ببعض مهام إدارة قواعد البيانات الأخري .



شكل رقم (٣)

يلاحظ من الشكل السابق أن برامج الوحدة التابعة الخاصة بقاعدة البيانات تقوم بإرسال متطلبات قواعد البيانات الى وحدة خدمة خاصة بقاعدة البيانات حتى تقوم بدورها بمعالجتها.

٦/٢/١ وحدات الخدمة الخاصة بقاعدة البيانات :

يتم تشغيل وحدة خدمة الخاصة بقاعدة البيانات علي نظام الكمبيوتر الخاص بها وتستقبل متطلبات قاعدة البيانات عبر الشبكة . فإذا أسفرت المتطلبات عن وجود نتيجة ، يتم إظهارها داخل كمبيوتر الوحدة التابعة

الخاصة بقاعدة البيانات عبر الشبكة . ويؤدي هذا الترتيب إلي عزل وحدة الخدمة الخاصة بقاعدة البيانات بصورة مستقلة عن كل من برامج الوحدة التابعة الخاصة بقاعدة البيانات ، الأمر الذي من شأنه أن يعمل علي تحسين أداء وموثوقية نظام قواعد البيانات .

يتم تشغيل حزمة برامج معينة معروفة باسم (DBMS) Database Managemant System على وحدة الخدمة الخاصة بقاعدة البيانات . إذ تمثل تلك الحزمة البرنامج الذى من شأنه أن يستقبل مطالب قواعد البيانات ويقوم بمعالجتها وإظهار المعلومات الناتجة مرة ثانية إلى الوحدة التابعة الخاصة بقاعدة بيانات .

غير أن وحدة الخدمة الخاصة بقاعدة البيانات يتم تشغيلها على نظام تشغيل تم تصميمه لدعم وحدات الخدمة مثل Windows 2000 / NT أو أى من الإصدارت المختلفة من Unix . ويعزو السبب فى ذلك إلى أن نظم التشغيل المكتبية مثل Windows 98/95 لا تتوفر بها الموثوقية والإستقرار اللازمين لتشغيل وحدة الخدمة الخاصة بقاعدة البيانات .

فى البيئات الصغيرة ، يمكنك تشغيل عدة وحدات خدمة مختلفة على الكمبيوتر الخاص بوحدة الخدمة مثل وحدة خدمة ملف ووحدة خدمة الويب ووحدة خدمة البريد الإلكترونى ووحدة خدمة العمليات ، هذا فضلا عن وحدة الخدمة الخاصة بقاعدة البيانات . بيد أنه كلما زاد مقدار التحميل كلما كان من الأفضل أن يتم تخصيص أجهزة كمبيوتر لكل من هذه العمليات على نحو مستقل .

ويسمح لك ذلك النوع من الترتيب بقياس سعة كل نظام كمبيوتر حتى يتوافق مع مقدار التحميل . ويمكنك إضافة ذاكرة أخرى ووحدات CPU أسرع إلى جانب المزيد من الأقراص كى يكون هناك حيز كافى فى ظل تزايد مقدار التحميل . بالإضافة إلى ذلك عندما يتم تخصيص كمبيوتر كى يتم من خلاله تشغيل وحدة الخدمة الخاصة بقاعدة البيانات ، يقل خطر تدمير النظام

حيال ظهور مشكلة فـي برنـامج يوجد بوحـدة خدمـة أخـرى كوحـدة خدمـة المعاملات على سبيل المثـال .

٢/٢/٧ الوحدات التابعة الخاصـة بقـاعدة البيـانـات:

تعد الوحـدة التابعـة الخاصـة بقـاعدة البيانـات ببسـاطة بمثابـة برنـامج تطبيقى يتم تشغيله علـى ذات الشـبكة كمـا هـو الحـال مـع وحـدة الخدمـة الخاصة بقاعدة البيانات ، والذى مـن شـأنه أن يطلـب معلومـات مـن قـاعدة البيانـات . ويقـوم البرنـامج التطبيقـى بإنشـاء متطلبـات قـاعدة البيانـات باسـتخدام Applications Programming Interface (API) ، وهـى ليسـت أكثر من مجرد مجموعة من إسـتدعاءات السـبرامج الفرعيـة أو مجموعـة مـن العنـاصر يسـتخدمها البرنـامج لإرسـال المتطلبـات إلـى قـاعدة البيانـات وإستقبال المعلومات من قـاعدة البيانـات .

فضلا عن ذلك تسمح واجهة API للبرنـامج بإمكانيـة الإتصـال بـأكثر مـن نظام قاعدة بيانات من خلال إسـتخدام جـزء معيـن مـن الكـود يعـرف بإسـم برنامج التشغيل الخـاص بقـاعدة البيانـات ويمثـل برنـامج التشـغيل الخـاص بقاعدة البيانات برنامجا خاصـا يتـم تنصيبـه علـى كمبيوتـر الوحـدة التابعـة الخاصة بقاعدة البيانـات ، حيـث يـترجم مطـالب قـاعدة البيانـات المعياريـة بإستخدام API إلى لغة خاصة مسـتخدمة مـن قبـل وحـدة الخدمـة الخاصـة المتعلقة بقاعدة البيانـات .

ومن خلال إسـتخدام برامـج التشـغيل الخاصـة بقواعـد البيانـات ، يمكـن لنفس البرنامج التطبيقى الإتصال بنظـم مختلفـة لقواعـد البيانـات دون تغيـير التطبيق ذاته . ويعد هذا المستوى من الإسـتقلالية مـهما ، حيـث أنـه يسـمح للمنظمة بإستبدال نظام قـاعدة بيانـات بـآخر دون حـدوث تـأثير كبـير علـى التطبيقات ذاتـها .

٣/٦ نظم قواعد البيانات البسيطة :

رغم أن معظم نظم قواعد البيانات تعمل من خلال بنية الوحدة التابعة / وحدة الخدمة المذكورة أعلاه ، إلا أن هناك عدة نظم قواعد بيانات تعمل بصورة مختلفة . تلك هى نظم قواعد البيانات البسيطة ، حيث لا تمثل مسائل الأداء وقابلية التدرج فى قاعدة البيانات ثمة مشكلة . غير أن هذه النظم مازالت تتصل بالتطبيق من خلال واجهة API المعيارية ، لكن بدلا من تمرير المتطلبات داخل وحدة الخدمة الخاصة بقاعدة البيانات على كمبيوتر مختلف ، تقوم هذه النظم بتمرير المتطلبات إلى الكود الموجود فى الكمبيوتر الخاص بالوحدة التابعة .

من ناحية ، يتم تصميم الكود الخاص بقاعدة البيانات الموجودة فى الكمبيوتر المحلى المتعلق بالوحدة التابعة بحيث يتمكن من التعامل مع النسخ الأخرى من كود قاعدة البيانات والتى يتم تشغيلها فى أجهزة الكمبيوتر الأخرى بغرض الوصول إلى ملف شائع أو مجموعة ملفات موجودة فى موقع على الشبكة . ويمكن تحديد موقع الملفات الخاصة بقاعدة البيانات على نفس الكمبيوتر كواحدة من الوحدات التابعة الخاصة بقاعدة البيانات أو يتم تحديدها كوحدة خدمة ملفات مركزية . فضلا عما سبق ، يعد نظام قاعدة البيانات البسيط كذلك إختيار صائبا فى حالة ما إذا كنت تقوم بتصميم تطبيق مستقل لمستخدم واحد .

بالإضافة إلى ذلك ، تعد إدارة نظم قواعد البيانات البسيطة تلك سهلة للغاية. إذ يتم تنفيذ عملية النسخ الإحتياطى لقاعدة البيانات من خلال إعداد نسخة إحتياطية من الملف الذى يحتوى على قاعدة البيانات ، ومن ثم يتم حصر عمليات الإسترداد على إسترجاع نسخة إحتياطية من قاعدة البيانات .

مما لاشك فيه أن ذلك المنهج سيكون لدية تأثير كبير على الأداء ، نظرا لأن التأمين ينبغى القيام به الآن على مستوى الملفات بدلا من القيم التى تم تحديدها من خلال قاعدة البيانات . ورغم أنه من الواضح أن ذلك يقلل من مستوى فعالية التطبيق الذى يستعمل من قبل عدة مستخدمين ، إلا أن حمل

الأداء فى هذه النظـم يكـون ضئيـلا بدرجـة كافيـة بحيـث يسـتوعب عـددا مـن المستخدمين لا يتجاوز أصابع اليد ، ومن ثم قـد يكـون الأداء بالنسـبة لـهم أفضـل بالفعل من بنية الوحدة التابعة / وحـدة الخدمـة .

(٧) برمجيـات المنـافع لقواعـد البيانـات :

من أجل مساعدة المتعاملين مـع قـاعدة البيانـات بـإختلاف مسـتوياتهم علـى تحقيق متطلباتهم والقيام بأعمالـهم ، فقـد وضعـت العديـد مـن الـبرامج الفائديـة Utilities منـها :

- **اللغات الإستفسارية** : تصمـم اللغـات الإستفسـارية Query Languages مـن أجـل المستخدمين لكى تمكنهم من القيام بإستفسـاراتهم مـن القـاعدة . وبعكـس أوامـر اللغـات التى تتعامل مع البيانات مثل الكوبول ، فإن هـذه اللغـات تمكـن مـن توجيـه إستفسـارات للقاعدة وفق متطلبات المستفسر ، وهـى تتمـيز بكونـها سـهلة الفـهم والتطبيـق لكـون مصطلحاتها قريبة للغاية من اللغة العاديـة ، فمثـلا الأمـر :

Display All Employee, Employee_name .Form Empipyee. Where Employee_age >59

يستفسر من قاعدة البيانـات عـن كافـة الموظفيـن المتوقـع خروجـهم على المعاش هذا العام . ومن المسـتحيل عمليـا الحصـول علـى مثـل هـذه الإستفسارات من النظام المؤسس علـى الملفـات .

- **قواميس البيانات** : يمكن تعريـف قواميـس البيانـات بأنـها مخـزن للبيانـات عن البيانـات الموجـودة القـاعدة . فـهذا القـاموس يضـم أنـواع البيانـات وأسماء السجلات وهياكلها ومعلومات أخـرى عـن القـاعدة .

- **أدوات المحاسبة والرقابة** : وتسـتخدم لتحديـد مـدى إسـتخدام قواعـد البيانـات من قبل الأفراد والإدارات ومراكـز التكلفـة .

- **مولدات التقارير** : قد يكون من الضرورى أحيانـا تقديـم مخرجـات البيانـات فى أشكال معينـة لزيـادة توضيـح المعلومـات ، وفـى المقـابل قـد يكـون إظهارها فى صورة قياسية تقليديـة ، كالميزانيـات . وبينمـا تخـرج التقـارير

من قاعدة البيانات في صورة قياسية ، فإن مولدات التقارير تسمح بإخراجها في صورة مختلفة وهي أدوات قوية وسهلة الإستخدام .

- **النسخ الإحتياطية وإسترجاع البيانات المفقودة** : من المعتاد تخزين قواعد البيانات على نسخ إحتياطية من وقت لآخر وذلك لإستخدامها في حالات حدوث أي طارئ للقاعدة ، [تسمى عملية النسخ الكلي للبيانات Dumping] وفيما بين عمليات النسخ يجري تسجيل أية معاملة تؤثر في القاعدة (حذف ، إضافة ... إلخ) ، وتسمح البيانات المخزنة إحتياطيا مع العمليات المسجلة لإستعادة القاعدة بعد حدوث أعطال بها .

- **التحكم في تزامن العمل** : يمكن أن يتقاسم أكثر من مستخدم أو تطبيق العمل على نفس البيانات ، وليس من مشكلة إذا كان المطلوب هو مجرد قراءة البيانات ، ولكن المشكلة تثور إذا حاول أكثر من مستخدم تعديل بيان ما في نفس الوقت ، وتتكفل نظم إدارة قواعد البيانات مواجهة هذا الغرض بـ " إغلاق الطريق " أمام أحد المستخدمين حتى ينتهي الآخر من عمليته .

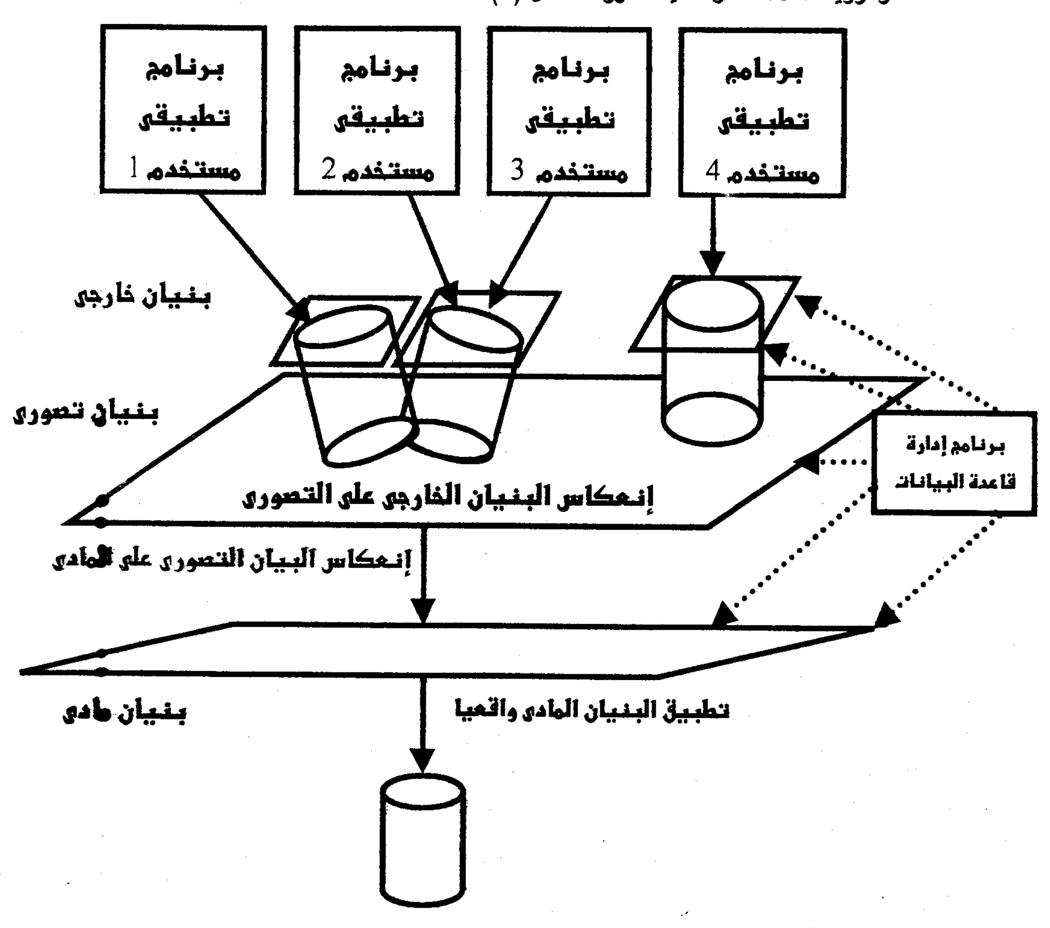
- **إعادة التنظيم المادي** : ويساعد ذلك مشرف القاعدة في إعادة بنائها إذا دعت الحاجة لذلك حيث أن العمليات المتكررة على القاعدة من حذف وإضافة وتعديل تسبب إبطاء العمل تدريجيا ، فيما يتطلب إعادة تنظيم التخزين بين الحين والآخر .

(٨) **المستويات الثلاثية لهيكل قواعد البيانات** :

تم بلورة بعض الأسباب الأساسية لإستخدام قواعد البيانات من حيث :

- فصل البيانات عن التطبيقات التي تستخدمها .
- عرض البيانات برؤية منطقية بصرف النظر عن التفاصيل المادية للتخزين .
- الإقتصار في تقديم البيانات على ما يهم المستخدم أو التطبيق منها .

من المفاهيم الجوهريــة لفــهم نظـم قواعــد البيانـــات مفــهومى البنيـــان Schema والرؤية View هو ما يصــوره شــكل (٤) .



شكل (٤) المستويات الثلاثة لهيكل قاعدة البيانات

وبصــورة عامــة فــإن البنيــان التصــورى Conceptual Schema هــو الرؤيــة المنطقيــة ، أو الذهنيــة لقــاعدة البيانـــات بأكملــها ، والبيـــان الخـــارجى External Schema فــهو رؤيـــة التطبيقـــات المختلفـــة كـــل بطريقتــه لقـــاعدة البيانـــات ، أن البنيـــان المـــادى أو الداخلى Internal Schema فهو الرؤية الماديـــة أو الواقعيـــة لقـــاعدة البيانـــات والتـــى تقـــدم معلومات حول تخزين البيانات ماديا والتى لاشأن بالرؤيـــة المنطقيـــة للبيانـــات بـــها .

البنيـان التصـورى :

هــو الرؤيـــة المنطقيـــة للبيانـــات ، فـــهو يحـــوى ، ضمـــن تفـــاصيل أخـــرى ، مواصفات عـــن :

- نوعية البيانات المحفوظة عن كل نوعية مـــن الكينونـــات التـــى تضمـــها القـــاعدة ، مثال ذلك تكون البيانات المحفوظة عـــن الموظـــف كـــالآتى :

  رقم الموظف : حقل رقمى ، طوله ســـتة خانـــات :

  تاريخ الميلاد : حقل تـــاريخى :

- العلاقات بين الكينونات الداخلة فى القاعدة ، مثـــل : المـــورد يـــورد البضاعـــة .

  أية تحديدات للبيانات ، مثـــل : لايزيد رقـــم الموظف عـــن خمســـة أرقـــام .

- أكواد الترخيص للعاملين فى التعامل مع القـــاعدة ، مثـــل : مرتبـــات الموظفيـــن لا يقرأ إلا بواسطة حاملى أكواد التـــرخيص 5 ، 6 ، 9 ولاتـــعدل إلا عـــن طريـــق حامل الكـــود 9.

ويعرف البنيان التصورى من خلال لغة خاصـــة تســـمى " لغـــة تعريـــف البيانـــات Data Definition Language(DDL) " . وتعـــرف هـــذه اللغـــة فـــى قـــاعدة البيانـــــات المستخدمة . ويمكن أن نعتبر أن هذا البنيان منبثقا مـــن نمـــوذج للمنطقـــة ، لـــذا يجـــب تصميمه بعناية ، حيث أن هيكلة يظل ثابتا بـــلا تغيـــير .

البنيـان الخـارجى :

كل مستخدم أو تطبيق لا يهمه مـــن قـــاعدة البيانـــات إلا القـــدر الـــذى يتعـــامل معه ، ومن ثم تكون له رؤيته الخاصة ، ومن ثـــم فـــإن البنيـــان التصـــورى يتفـــرع إلـــى عدد من الأبنية الخارجية ، كل بنيان خارجى يمثل رؤيـــة مســـتخدم أو تطبيـــق معيـــن .

فقد لا تتطلب طبيعة أحد المستخدمين مثـــلا إلا أنواعـــا معينـــة مـــن الســـجلات ، ومن خلال هذه السجلات قـــد لايحتـــاج إلا لعـــدد مـــن الحقـــول [بينمـــا يـــهتم موظـــف التأمينات بعمر الموظف ، فموظف الأجـــور لايهمـــه هـــذا البيـــان] ، وهنـــا يقـــدم كـــل بنيان خارجى نافذة خاصة بكل مستخدم أو تطبيـــق علـــى البنيـــان التصـــورى .

البنيان المادي :

يصف البنيان المادى كيفية تنفيذ قاعدة البيانات بالفعل ، فيهتم بطرق التخزين ، وأنظمتها ، وطرق الوصول للبيانات كالفهرسة وحقول المؤشرات والصيغ الرياضية لحساب العناوين ، وعموما كل مما لايدخل فى نطاق البنيان التصورى .

العلاقات بين الأبنية المختلفة :

بالإضافة إلى الحفاظ على هذه الرؤى ، تحتاج نظم إدارة قواعد البيانات كذلك إلى معرفة كيفية إرتباط كل رؤية بالأخرى ، بحيث أنه حين يعدل البنيان المادى مثلا ، لا يتأثر البنيان التصورى، وحين يعدل البنيان التصورى بإضافة خصيصة جديدة لسجل ، (كالحالة الصحية للموظف) ، لا يؤثر ذلك على الأبنية الخارجية الخاصة بالتطبيقات التى لا تتعامل مع هذه الخصيصة .

(٩) أنواع قواعد البيانات :

هناك أربعة أنواع رئيسية من نظم قواعد البيانات : المتسلسلة والشبكية والمفهرسة والإرتباطية . ورغم وجود مفاهيم مشتركة بين كل من الأنواع الأربعة ، إلا أن الإختلافات تنشأ إزاء الأسلوب الذى يتبعه كل منها فى تخزين البيانات .

١/٩ قواعد البيانات التسلسلية :

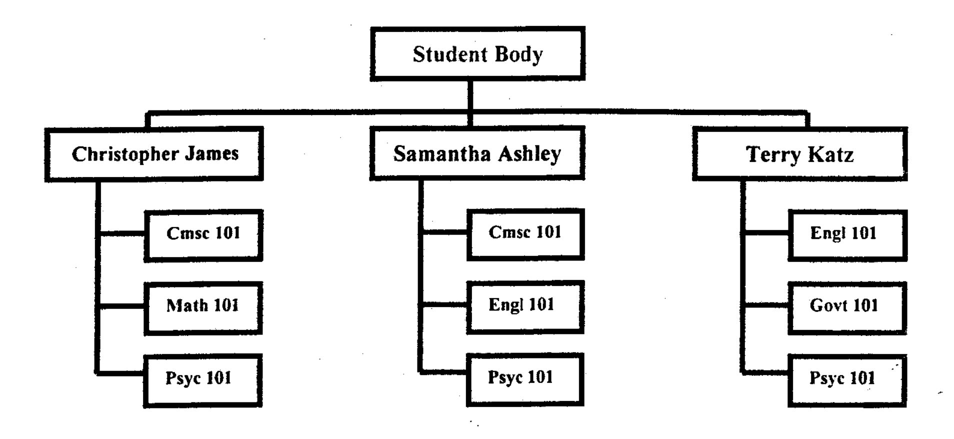
تمثل قاعدة البيانات التسلسلية أولى أشكال قواعد البيانات . إذ يتم ترتيب البيانات على هيئة سلسلة من البنيات التفصيلية ، الأمر الذى غالبا ما يعكس العلاقة الطبيعية بين البيانات . وتعد قاعدة بيانات IMS الخاصة بشركة IBM – التى مازالت متاحة حتى الآن – مثالا نموذجيا على قاعدة البيانات التسلسلية .

ويشكل التسلسل الهرمى العلاقة بين البيانات فى ضوء العلاقة المعروفة بإسم علاقة رأس بأطراف . ويعنى مصطلح علاقة طرف بأطراف

أنه يتم ربط إحدى قيم البيانات بواحـــدة أو أكــثر مــن قيــم البيانـــات الأخــرى . ويتمثل المثال النموذجى لهذا النوع من العلاقات فـــى الطلبــة والمــواد الدراســية . عندما يحضــر الطلبــة إلــى المدرســة ، فإنــهم يتلقــون العديــد مــن المــواد الدراسية ، وهذه هى إحدى الحالات التـــى يمكــن توضيحــها مــن خــلال علاقــة تسلسلية ، وذلك على النحو الموضوح فـــى شــكل (٥) .

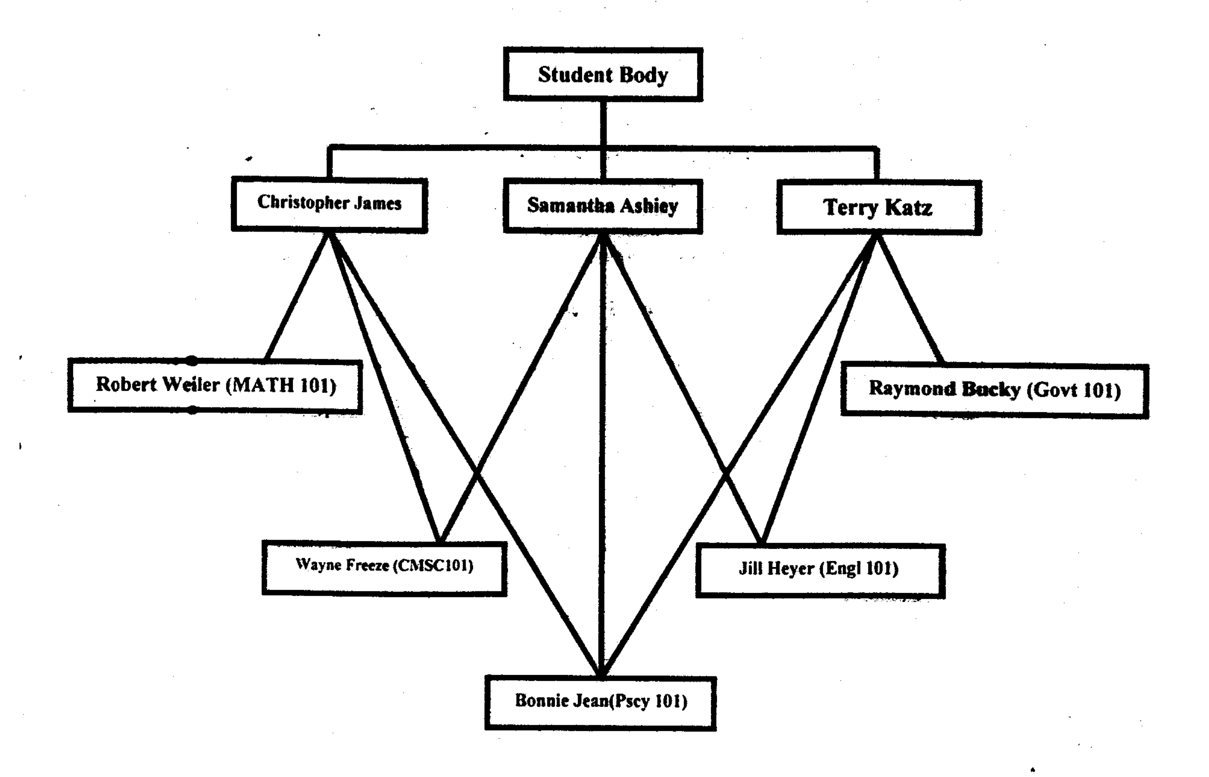
٢/٩ قواعـد البيانـات الشـبكية :

تم تطوير نمط قـــاعدة البيانـــات الشـــبكية كـــى يضحـــى بديـــلا لنمـــط قواعـــد البيانات التسلسلية . فرغم أنه يمكـــن تنظيـــم بيانـــات كثـــيرة باســـتخدام العلاقـــة التسلسلية ، إلا أنه من الصعب أن يتم تشـــكيل علاقـــات أخــرى بيــن البيانـــات . ضع فى إعتبارك مثال الطلبة والمدرسين : فالطـــــالب ســـوف يتلقـــى العديـــد مـــن الدورات عن طريق عدة مدرســـين ، كذلـــك الحـــال مـــع المـــدرس فســـوف يعلـــم أكثر من طالب ، وذلك على النحو الموضح فـــى شـــكل (٦) . ويطلـــق علـــى هـــذه العلاقة مصطلح علاقة أطـــراف بـــأطراف .



الشكل (٥)

تعكس قاعدة البيانات التسلسلية العلاقة الطبيعية بين البيانات



الشكل (٦)

يمكن القاعدة البيانات الشبكية أن تعالج حالات يتعذر معالجتها من خلال قاعدة البيانات التسلسلية

تقوم قاعدة البيانات الشبكية بتخزين المعلومات فى صورة مجموعات بيانات تشـبه فى هيئتها الملفات والجداول . وذلك نظرا لأن السجل الموجود بمجموعة بيانات والمرتبـط بمجموعة من السجلات فى مجموعة بيانات أخرى سيكون لدية مجموعة مـن المؤشـرات الفعلية تعرف أيضا بإسم الرابط . من ثم ، يرتبط السجل الموجود فـى مجموعـة البيانـات الأولى بالسجلات الموجودة فى مجموعة البيانات الثانية . ويتمثل العـائق الأساسـى لـدى قواعد البيانات الشبكية فى عملية الإحتفاظ بجميع الروابط ، بإعتبارها أمر معقد للغايـة فـإن وجود رابط واحد غير فعال ينجم عنه العديد من المشاكل فى قواعد البيانات .

٣/٩ قواعد البيانات المفهرسة :

رغم أن قواعد البيانات المفهرسة تشبه قواعد البيانات الشبكية من حيث المفهوم ، إلا أن قواعد البيانات المفهرسة تستبدل الروابط بمجموعة من القيمة المميزة . وتشكل مجموعة القيم تلك ملفا آخر وتوجد فى حقل معين بإحدى مجموعات البيانات ، هذا فضلا عن وجود مؤشرات لكل سجل تظهر فيه القيمة . وتعرف هذه القيمة بأنها قيمة أساسية . فإذا علمت القيمة الأساسية ، ستتمكن من سرد جميع السجلات التى تحتوى على هذه القيمة فى مجموعة البيانات . ومن الجدير بالذكر أن قاعدة البيانات المفهرسة تعتبر بمثابة أكثر نظم قواعد البيانات فعالية فى هذه الآونة .

٤/١٠ قواعد البيانات الإرتباطية :

يرجع وجود مفهوم نظام إدارة قواعد البيانات الإرتباطية (RDBMS) إلى أوائل السبعينات ، حينما كان باحثو IBM يحاولون كشف النقاب عن أفضل وسيلة لإدارة البيانات . إذا تتطلب قواعد البيانات التسلسلية والشبكية والمفهرسة جميعا مؤشرات لكى يتم ربط البيانات معا ، مما يعتبر إحدى المشاكل التى يمكن مواجهتها عند إستخدامها . لذلك ، أنشأ الباحثون نمط رياضى طرح المؤشرات جانبا وإستعان بالقيم الشائعة لربط السجلات المتعددة معا . من ثم ، لم يقتصر الأمر على البيانات التسلسلية كما هو الحال فى نمط قواعد البيانات التسلسلية ، كما إنك لم تكن مضطروا لضبط الروابط مثلما يحدث فى نمط قواعد البيانات الشبكية .(مع الوضع فى الإعتبار أن نمط قواعد البيانات المفهرسة ظهر إلى حيز الوجود بعد إنتهاء مرحلة العمل الأساسية فى نمط قواعد البيانات الإرتباطية) .

من ناحية أخرى ، تتمثل المشكلة الأساسية لنمط قواعد البيانات الإرتباطية فى كونها بطيئة للغاية بالمقارنة مع أنماط قواعد البيانات الأخرى . فمع عدم وجود مؤشرات تساعدك على تحديد موقع سجل معين على نحو سريع ، قد تضطر إلى قراءة جميع السجلات الموجودة فى الجدول للعثور على سجل واحد . وإسهاما فى حل هذه المشكلة ، تم تقديم مفهوم الفهرس . غير أن قواعد البيانات الارتباطية تعتبر أقل فاعلية بوجة عام من قواعد البيانات التى يتم تنفيذها من خلال أنماط

قواعد البيانات الأخرى . وذلك حتى فى ظل وجود الفهارس . إلا أنه وعلى الرغم من ذلك ، ذاع صيت قواعد البيانات الإرتباطية عندما تطور مجال أجهزة الحاسبات من أجهزة الحاسبات الإلكترونية الكبيرة التقليدية إلى أجهزة الكمبيوتر الصغيرة وأجهزة الكمبيوتر الشخصية . بيد أنه لم تكن التطبيقات المبدئية الخاصة بقواعد البيانات الإرتباطية مطلوبة بصورة كبيرة بالمقارنة مع نظم قواعد البيانات التى يتم تشغيلها على الحاسبات الكبيرة . لذا فلم يمثل عامل الأداء ، عاملا مهما كما ينبغى أن يكون . وبمرور الوقت زادت سرعة أجهزة الكمبيوتر الصغيرة والشخصية إلى الدرجة التى أضحت عندها تتنافس مع الحاسبات الكبيرة .

(١٠) قواعد البيانات الشائعة الإستخدام :

هناك عدد من نظم قواعد البيانات الارتباطية التي يمكنك الاستعانة بها والمتاحة حاليا في الأسواق . وسوف نتناول في هذا الكتاب ثلاثة فقط من هذه النظم بصورة مركزة ألا وهي : Microsoft SQL Server 7.0 و Microsoft Jet 3.5/4.0 و Oracle 8i تمثل نظم قواعد البيانات الثلاثة تلك أغلبية نظم قواعد البيانات المستخدمة في العالم اليوم ، علي الأقل من وجهة نظر مبرمج Visual Basic . ورغم أن هناك نظم أخري من قواعد البيانات التي يتم تشغيلها في عالم الحاسبات الكبيرة – مثل CA's IDMS و IBM's IMS – إلا أنها غير مستخدمة من قبل مبرمجي Visual Basic .

١/١٠ نظام قاعدة بيانات SQL Server 7.0

يتمثل أحد أهداف شركة Microsoft الرئيسية وراء تصميم نظام قاعدة بيانات SQL Server 7.0 في إنشاء منتج ذي قابلية علي التدرج بالنسبة لقواعد البيانات . بحيث يتم تشغيله بنفس الكفاءة علي كل من نظام Windows 98 البسيط ذي الذاكرة المحدودة إلي جانب مساحة القرص المتاح ونظام Windows 2000/NT الضخم ذي المعالجات المتعددة والذاكرة الرئيسية التي يتم قياس السعة الخاصة بها بوحدات الجيجابايت فضلا عن مساحة كبيرة من القرص . لإنجاز هذه المهمة ، قامت شركة Microsoft بإنتقاء ثلاثة إصدارات من SQL Server . يتمثل الإصدار الأول في Desktop Edition ويتم تشغيله علي Windows 98 و Windows 2000/NT

Workstation . وقد تم تصميم هذا الإصدار لمعالجة قواعد البيانات الأصغر ، وهو يتسم بكونه إصدار مثالي فيما يتعلق بمساعدة المطورين على اختبار البرامج الخاصة بهم بعيدا عن وحدة خدمة الإنتاج الخاصة بقاعدة البيانات . بينما يمثل Standard Edition أكثر الإصدارات شيوعا حيث يتيح السمات التي أنت بحاجة إليها بالفعل بالنسبة لمعظم البرامج التطبيقية . أما Enterprise Edition فهو إصدار متطور عن Standard Edition نتيجة لإضافة سمات تساعد SQL Server على نعالجة البرامج التطبيقية التي تحتوي علي بيانات كثيرة وعدد ضخم من العمليات .

فضلا عن كونها تركز علي قابلية التدرج بالنسبة لقواعد البيانات ، فإن شركة Microsoft كذلك بذلت جهودا مضنية للتقليل من التكاليف الإجمالية لامتلاك نظام قواعد البيانات . إذ أنه لايتسني تشغيل قواعد البيانات الارتباطية الحديثة إلا عن طريق مجموعة من المتخصصين يطلق عليهم مسؤولي قواعد البيانات . غير أن الاستعانة بمثل هؤلاء مسألة تتكلف أجورا باهظة ، هذا فضلا عن أنه من الصعب الاحتفاظ بهم في ظل الطلب المتزايد علي ذوي هذه المهارات . ويعالج نظام قاعدة بيانات SQL Server 7.0 هذه المشكلة من خلال وضع عدد من المعالجات التي تجعل من القيام بالمهام الروتينية أمرا سهلا . بالإضافة إلي ذلك ، يمكن استخدام SQL Server Agent لتشغيل فعاليات متعددة عند عدم استخدام الجهاز .

بالإضافة إلي ذلك ، تناولت كذلك شركة Microsoft التطبيقات المتعلقة بتخزين البيانات من خلال وضع سمتين رئيسيتين في نظام قاعدة بيانات SQL Server 7.0 . تتمثل الأولي في Data Transformation Services (DTS) . وتسهل هذه السمة من انتقال البيانات من مكان إلي آخر . ويعد ذلك أمرا ذا أهمية كبيرة ، لاسيما وأن البيانات الموجودة في مخزن البيانات عادة ما يتم استدعاؤها من برامج تطبيقية ذات غرض معين مثل النظم الخاصة بالحاسبات والجرد ، وذلك حتى إذا كانت موجودة في الأجهزة الرئيسية المشتركة .

أما السمة الأساسية الثانية فتتمثل في OLAP Services (وتعد OLAP اختصارا لمصطلح Online Analytical Processing). وتعتبر هذه الخدمات بمثابة حلقة الوصل بين مخزن البيانات وأدوات التحليل التي يتم تشغيلها على محطة العمل

الخاصة بالمستخدم . إذ يتم معالجة البيانات الموجودة في مخزن البيانات مسبقا من خلال وحدة خدمة OLAP قبل تقديم النتائج إلي أداة التحليل . ولقد جعلت Microsoft واجهة البرمجة التطبيقية (API) متاحة بالنسبة لـ OLAP Services ، من ثم يمكن لشركات أخري أن تتمكن من تصميم الأدوات الخاصة بالوحدة التابعة حتى يتسني لها تحليل بيانات OLAP.

١٠/٢ نظام قاعدة بيانات Microsoft Jet 3.5/4.0

يعد برنامج Microsoft Access بمثابة أداة تطوير مكتبة بسيطة خاصة بتطوير قواعد البيانات وقد تم إنشاؤه خصيصا من أجل المنظمات ذات النشاطات المحدودة والمتوسطة . ويوجد في الجزء الأساسي من برنامج Access آلية قاعدة بيانات ارتباطية حقيقية تعرف باسم Microsoft Jet . وكما هو الحال مع نظام قواعد بيانات SQL Server ، يعتمد نظام قواعد بيانات Jet علي لغة SQL القياسية . ولكن علي العكس من نظام SQL Server ، يتم تشغيل الكود الخاص بقاعدة البيانات في التطبيق نفسه وليس في وحدة الخدمة الخاصة بقاعدة البيانات . بالإضافة إلي ذلك ، نظرا لأنه تم تطوير نظام Jet بصورة مستقلة عن SQL Server، فإنه ليس متوافقا مع نظام SQL Server علي نحو تام .

ورغم أنه يتم تطوير Jet بمثابة جزء من برنامج Microsoft Access ، إلا أنه يأتي مرفقا مع Visual Basic ، إذ يمكنك تطوير التطبيقات باستخدام Jet ، كما يمكنك كذلك توزيع الكود الفعلي الخاص بنظام Jet كيفما تشاء من خلال التطبيق ، الأمر الذي لا يمكنك القيام به مع تطبيقات SQL Server .

يعد Jet 3.5 إصدار Jet المرفق ببرنامج Access 97 و Visual Basic 6 ، أما Jet 4.0 فهو الإصدار المرفق ببرنامج Access 2000 ، رغم أن Jet 4.0 يتمتع بوجود بعض المميزات عن Jet 3.5 ، إلا أنه غالبا ما ينبغي مواصلة العمل مع Jet 3.5 ما لم تكن بحاجة إلي مشاركة البيانات مع تطبيق Access 2000 ، وذلك نظرا لأن Access 2000 يعمل فقط مع Jet 4.0.

١٠/٣ نظام قاعدة بيانات Oracle 8i

يعد Oracle 8i نظام قواعد بيانات عالى الكفاءة يمكن تشغيله على العديد من نظم التشغيل المختلفة . فرغم أن SQL Server متاح فقط للنظم المعتمدة أساسا على نظام تشغيل Windows ، إلا أن Oracle 8i يتسنى تشغيله على أى نظام تشغيل بدء من نظم Linux الصغيرة ووصولا إلى النظم التى تعتمد بصورة أساسية على نظام تشغيل Sun Solaris وأجهزة الكمبيوتر الرئيسية IBM التى يتم تشغيلها على MVS/ESA أو OS/390 . وبالطبع يمكن تشغيله كذلك على Windows 2000/NT . ونظرا لإستخدام نفس الكود مع جميع بيئات التشغيل على إختلاف أنواعها ، فإن نوع الكمبيوتر المضيف لوحدة الخدمة الخاصة بقواعد البيانات لايمثل ثمة مشكلة بالنسبة لتطبيقات Oracle 8i.

وكما هو الحال مع SQL Server ، فإن Oracle 8i متاح فى إصدارات متعددة ويمثل إصدار Standard Edition أكثر الإصدارات شيوعا من Oracle 8i ، حيث أنه يتوافق مع معظم تطبيقات قواعد البيانات . أما إصدار Enterprise Edition من Oracle 8i فقد تم تصميمه لدعم معالجة المعاملات الفورية (OLTP) ذات الحجم الكبير فضلا عن مخازن البيانات كثيرة الإستعلامات . بينما يتم تخصيص إصدار Personal Edition من Oracle 8i للتعامل مع تطبيقات التطوير ذات المستخدم الواحد .

خلاصة القول - أنه فى ضوء ما تقدم عرضه نجد أن قواعد البيانات ليست مجرد مدخلات ومخرجات بل هى " أحد ثروات المنظمة تحتاج لحكمة فى التنظيم والإدارة " .

وقاعدة البيانات هى مخزن للبيانات يمكن إستخدامها فى تطبيقات متعددة ، ويجب تصميمها بحيث تخدم إحتياجات هذه التطبيقات الحالية والمستقبلية ، خاصة لإستخلاص المعلومات التى تتطلبها إحتياجات المنظمة ، وبحيث يتم ذلك بطريقة مرنة تساعد فى عملية صنع القرارات . ولهذا السبب تعتبر قواعد البيانات نواة نظام المعلوماتية الشامل والمتطور.

والمميزات الأساسية لقاعدة البيانات الحديثة هى :

- أنها مخزن متكامل للبيانات المشتركة لخدمة متطلبات العديد من المستخدمين والتطبيقات .
- أنها مصممة بأسلوب ذى مغزى منطقى بالنسبة للمنظمة ، فمثلا إذا كانت هناك بيانات محفوظة عن الموظفين والمشاريع التى يعملون بها ، فستكون قاعدة البيانات شاملة للبيانات الخاصة بكل موظف وأيضا المشروع الذى يعمل به .
- عدم تكرار البيانات بقدر الإمكان .

وتحفظ قواعد البيانات جاهزة على الأقراص ، ومن الجوانب الهامة أن برمجيات نظام إدارة قواعد البيانات Database Management System (DBMS) تقع كواجهة بين المستخدمين وقاعدة البيانات ، حيث لا يتم التعامل مع البيانات إلا من خلال تلك البرمجيات وأهم ما يميز نظم إدارة قواعد البيانات .

تتعامل مع عمليات الوصول والكتابة والقراءة للبيانات التى يقوم بها المستخدمون والتطبيقات :

- تقدم للمستخدمين رؤية منطقية للأجزاء التى تهمهم من القاعدة .
- تخفى عن المستخدمين كيفية تخزين البيانات وعمليات إسترجاعها .
- تضمن ثبات قاعدة البيانات .
- تسمح للمستخدمين المختلفين بالدخول لقاعدة البيانات بحسب السلطات المخولة لهم .
- تسمح للمستخدم بتحديد هيكل القاعدة .
- تقدم إمكانيات متعددة للمراقبة والتحكم فى القاعدة .

الفصل الثانى

النماذج والأبنية فى بيئة قواعد البيانات

الفصل الثانى

النماذج والأبنية فى بيئة قواعد البيانات

سبق أن أوضحنا أن البيان التصورى ينبثق من أحد نماذج المنظمة ، ويجب توضيح أن البنيان التصورى يتحدد من خلال لغة تعريف البيانات الخاصة بنظام إدارة القاعدة المستخدم ، ويفرض كل نظام قيودا على ما يمكن أو لا يمكن تعريفه ، فهناك نظم متشددة فى مقابل دخول سريع للقاعدة ونظم مرنة على حساب هذه السرعة .

وتنقسم معظم نظم إدارة قواعد البيانات إلى ثلاث فئات من حيث القيود التى تفرضها ، وهذه الأنواع تقابل ثلاثة أنواع متميزة من نماذج هياكل البيانات وهى : النموذج الشبكى ، والنموذج الهرمى ، والنموذج العلائقى . فنموذج البيانات هو إذان نوع هيكل البيانات الأكثر ملاءمة لنظم إدارة قواعد البيانات ، والبنيان التصورى هو تعريف هذا النموذج من خلال لغة تعريف البيانات فى نظام إدارة قاعدة البيانات .

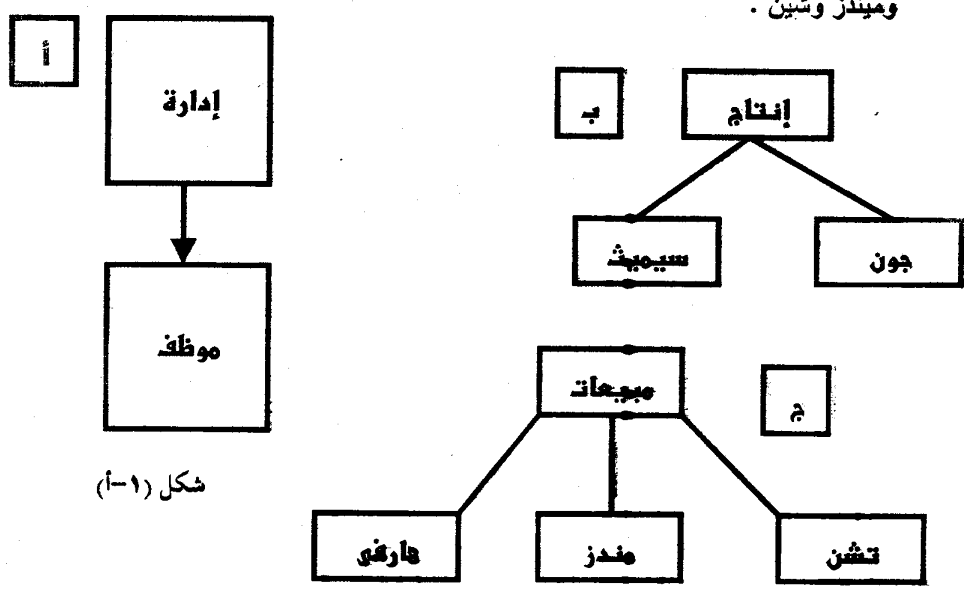
والأنواع الثلاثة لنماذج البيانات التى نناقشها هناك تعتبر هامة للغاية ، فقد كانت هى الفاعل المؤثر وراء برمجيات قواعد البيانات التجارية ، فهناك أنواع أخرى من النماذج لم يكن لها نفس التأثير.

(١) العلاقات والوقائع :

أحد الفروق الهامة بين الملف وقاعدة البيانات هو أن الأول عبارة عن مجرد تجميع لسجلات نفس النوع بينما تتكون الثانية من :

- أنواع مختلفة من السجلات .
- مجموعات مختلفة من هذه السجلات .
- روابط بين السجلات تعتمد على ما إذا كان هناك علاقات فى العالم الواقعى الكينونات التى تمثلها هذه السجلات .

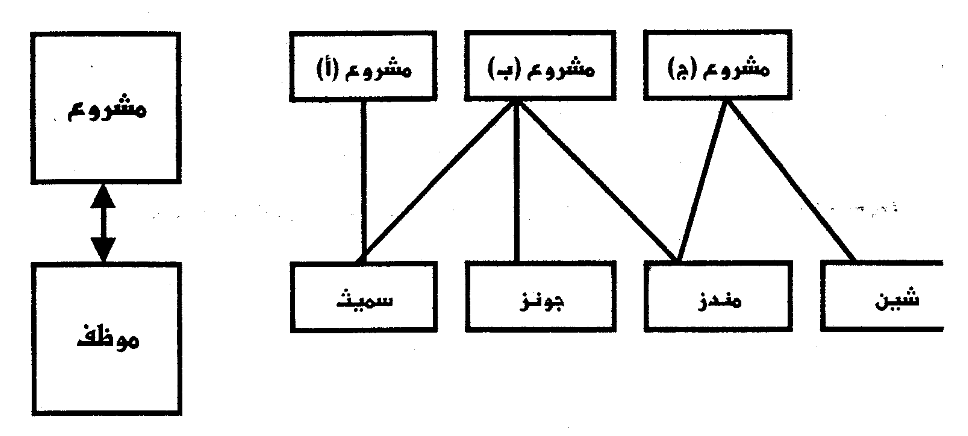
والعلاقات Relations بين الكينونات قد تكون من نـوع 1: ن 1:n relation ، أى " واحد إلى كثير " او ن : م n:m relation أى " قليل إلى كثير " . ويبين شكل (١-أ) تصويـر للعلاقة الأولى وهى بين الإدارة والموظف ، فكل إدارة تضم عدداً مـن الموظفيـن ، ولكـن لايمكن أن يتبع الموظف إلا إدارة واحدة ، ونلاحظ أن هذه العلاقة ممثلة بسهم يتجـه نحـو الكينونة المتعددة (الموظف) أما شكل (١-ب) فيمثل وقائع Occurrences (تطبيقات) لـهذه العلاقة ، فإدارة الإنتاج يتبعها كل من سميث وجونز ، وإدارة المبيعات يتبعها كل من هـارفى وميندز وشين .



شكل (١-أ)

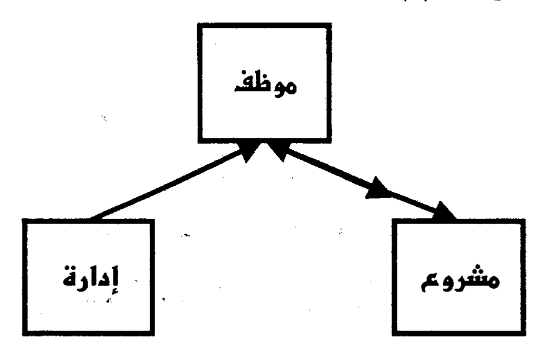
شكل (١-ب) يوضح علاقة من نوع 1 : ن) بعض وقائع العلاقة

بين شكل (٢-أ) تصويرا لعلاقة " قليل إلى كثـير " ، وهـى علاقـة المشـروعات بالموظفين ، فبينما يعمل فى كل مشروع العديد من الموظفين ، يمكن أن يعمل الموظف فـى أكثر من مشروع فى نفس الوقت وتلاحظ أن العلاقة ممثلة بسـهم ذو رأسـين فـى إتجـاه الكينونات الأقل (المشروعات) ورأس واحدة فى إتجاه الكينونات الأكثر (الموظف) . ويصـور شكل (٢- ب) وقائع لهذه العلاقة .



شكل (٢-أ) علاقة من نوع ن : م ب) بعض وقائع العلاقة

وسوف نعرض للنماذج الأساسية الثلاثة لقواعد البيانات فيما يلى ، وسوف نستخدم النموذج المبين فى شكل (٣) كأساس للدراسة .



شكل (٣) نموذج لعلاقة الإدارات / الموظفين / المشروعات

(٢) النموذج الشبكى :

يتيح النموذج الشبكى عرض جميع العلاقات 1 : ن . ويستخدم هذا النموذج بصفة خاصة فى أجهزة الحاسب الكبيرة Mainframs والمتوسطة Minicomputers وهى أكثر تعقيدا من النظم السابقة ، حيث تسمح تلك النظم بالعلاقات بين وحدات البيانات فى جميع الاتجاهات وذلك كما يتضح من الشكلين رقما (٤) ، (٥) .

ومن القيود الأساسية التى يفرضها النموذج الشبكى على قاعدة البيانات الخاصة بـه عدم السماح بعلاقات ن : م ولكى نفهم كيفية تنفيذ هذا التقييد نأخذ مثال علاقة المشروعات بالموظفين المذكور سابقا . فكما يبين شكل رقم (٤-أ) ، أدخلت كينونة جديدة سميت " المهمة assinment "تتجزأ بها العلاقة المباشرة بين الموظفين والمشروعات ، الى علاقتين من نوع 1:ن بين الموظفين والمهام ، وبين المشروعات والمهام ، فكل موظف يمكن أن توكل له أكثر من مهمة ، ولكن لايمكن أن توكل المهمة لأكثر من موظف ، وكل مشروع تخدمه أكثر من مهمة ، وأيضا لايمكن لمهمة تكون فى نطاق أكثر من مشروع . وتسمى سجلات هذه العلاقة سجلات رابطة Link recirds .

والنموذج الشبكى لقاعدة لبيانات يشتمل على المكونات التالية التى قد يشترك فى كثير منها النموذج المبنى على العلاقات أيضا :

أ- أداة التفسير التفاعلى : Interactive Definition Facility (IDF)

يسهل هذا المكون على تقديم سلسلة من العروض التى تظهر على الشاشة والتى يقود بعضها إلى آداء البعض الآخر . وعن طريقها يمكن للمستخدم من تفسير الخصائص المنطقية والطبيعية لمخطط قاعدة البيانات بفعالية كبيرة ، وتعتبر شاشات التفسير التفاعلى IDF سهلة الفهم والإستخدام لإجراء تصحيح الأخطاء بطريقة مريحة .

ب- لغة تداول البيانات : Data Manipulation Language(DML)

تستخدم عبارات لغة تداول البيانات التى قد تبنى على لغة الكوبول أو لغة البيزيك بواسطة المبرمج مع بعض عبارات اللغة المستخدمة بالفعل كالكوبول على سبيل المثال فى إجراء وتقسيم البيانات الخاصة ببرامج التطبيق التى تفرز وتوصل وتعدل البيانات فى قاعدة البيانات . كما تسمح هذه اللغة فى القيام ببعض وظائف الضبط أو الرقابة . وعادة تكتب برامج المصدر Source Programs بلغة تداول البيانات التى تجمع فى إطار إستخدم لغة تجميع الكوبول أو البيزيك كبرامج عادية بدون الحاجة إلى المرور على جهاز التجميع من قبل Prepass Compiler .

ج- نظام ضبط قاعدة البيانات : Database Control System (DBCS) :

يشرف هذا البرنامج الرقابى على كل الأنشطة المتصلة بنظام إدارة قاعدة البيانات كما يقوم بأداء عدة وظائف مثل :

- تقديم نظام التفاعل البينى مع نظام التشغيل والبرامج النفعية الثانوية لكى يمكن الإستفادة بالمصادر الأخرى المتوفرة فى قاعدة البيانات .
- رقابة العمليت المتزامنة Concurrent بإستخدام نفس قاعدة البيانات .
- حفظ سجلات الحالة Status Registers فى قاعدة البيانات لكى توصل نتائج تنفيذ عبارة تداول البيانات لبرنامج التطبيق .
- تقديم النظام البينى التفاعلى مع نظام التشغيل والبرامج النفعية الثانوية لكى يمكن الاستفادة بالمصادر الأخرى المتوفرة فى قاعدة البيانات .
- تأدية البرامج الروتينية المطلوبة لما يكلف به الملف المعين المحفوظ بقاعدة البيانات .
- تداول عمليات التدقيق أو الإختبار للوصول إلى الهدف أو الحل النهائى .

د- لغة تداول البيانات التفاعلية :

Interactive Data M anipulation Language (IDML)

يقدم هذا المكون آداة تفاعلية تساعد فى الوصول لقاعدة البيانات ، ويستخدم نفس العبارات والأفعال الرئيسية التى تتوفر بواسطة لغة تداول البيانوت . وتنفذ عبارات لغة التداول عند الإدخال على شاشات النهايات الطرفية وتوفر المزايا التالية :

- تقويم مخطط نظام إدارة قاعدة البيانات :
- تقرير المنطقية الأكثر قبولا لبرنامج التطبيق .
- عرض البيانات المستمدة من قاعدة البيانات وآداء أى تعديلات عليها.
- عرض محتويات قاموس بيانات قاعدة البيانات .
- عرض توقيتات تنفيذ عبارات لغة تداول البيانات .

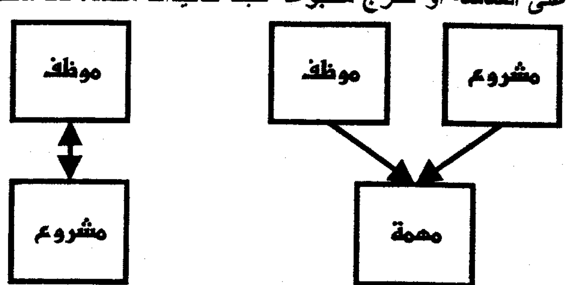
هـ- البرامج النفعية لقاعدة البيانات : D. 'base Utilities

يوفر نظام إدارة قاعدة البيانات عددا من البرامج النفعية التفاعلية التي يمكن أن تؤدى على شاشات النهايات الطرفية . وتقدم هذه البرامج النفعية التسهيلات التالية :

- إعداد وحفظ قاموس البيانات الذى يشتمل على خصائص قاعدة البيانات المنطقية والمادية ومسميات البيانات أو دلالاتها المفسرة من قبل المستخدم .
- المبادأة بإظهار ملفات قاعدة البيانات التى تشتمل على سجلات البيانات المطلوبة .
- طبع تقرير ملف البيانات لكى يساعد فى إستدعاء قاعدة البيانات .
- تحقيق وظيفة الإستدعاء فيما يتصل بقاعدة البيانات التى بها عطب Damaged .
- إعادة هيكلية قاعدة البيانات .

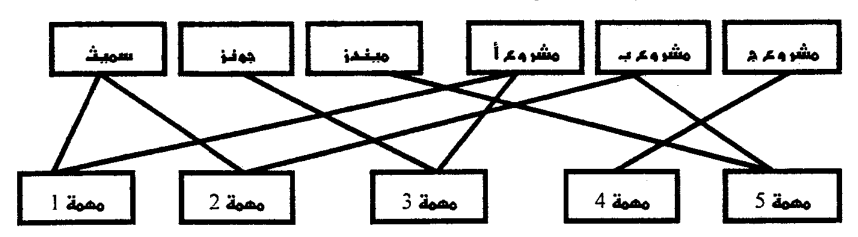
و- تصفح قاعدة البيانات Database Scan :

يسمح تسهيل التساؤل التفاعلى فى تصفح قاعدة البيانات بإدخال الأسئلة عشوائيا بدون الحاجة لكتابة برنامج تطبيق مستقل لها . وبذلك يوفر هذا المكون وسيلة تفاعلية ذات مستوى عال تسمح بإسترجاع تفسيرات أو مؤشرات السجل المتخصص أو البيانات الفعلية المختزنة فى سجل قاعدة البيانات عند مخاطبتها بواسطة المدخل المحدد لذلك فى قامو بيانات قاعدة البيانات . وتعرض البيانات إما على الشاشة أو تخرج مطبوعة طبقا لتأكيدات محددة تعد سلفا .



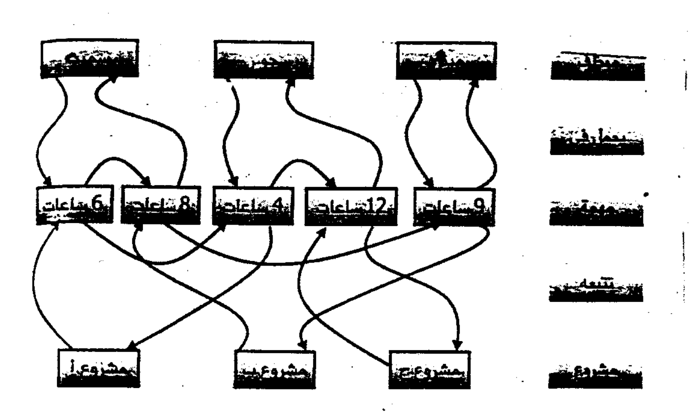
شكل (4) معالجة العلاقة م إلى : ن فى النموذج الشبكى

ويمكن أن تكون العلاقة المســتحدثة مجــرد علاقــة وهميــة لا تعنــى المـــنشأة فــى شئ [تســمى فــى هــذه الحالــة سجــلات وهميــة أو خــامدة Dummy records] ، ليست إلا حيلة يلجأ إليها لتنفيذ القيــد الــذى يفرضــه النمــوذج الشــبكى ، وتكــون الســجلات الرابطة خالية من البيانات ، كما يمكــن أن تكــون الكينونــة واقعيــة ، ولــها بيانــات فعليــة ، تحتوى سجلاتها الرابطة بيانات حقيقية ، كعدد الساعات التى إســتلزمتها كــها كـل مهمــة فــى مثالنا هذا . ويبين الشكل (٥) وقــائع للعلاقــات المذكــورة .



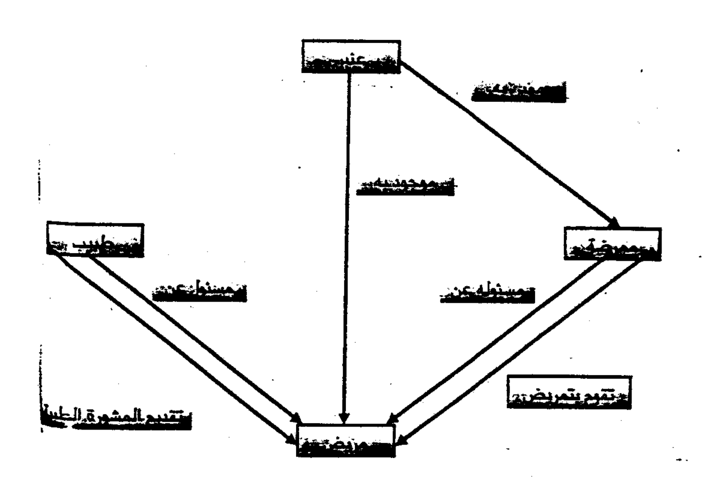
شكل (٥) وقائع لإستخدام سجل رابط

وتم الحفاظ على الروابط بين السجلات فى النموذج الشبكى بواســطة " المؤشــرات Pointers " ويصور شكل (٦) الهيكل المنطقى للمؤشرات بين الكينونات المختلفة ، ومنــها يتضح أن سجلات الربط الخاصة بالكينونة " المهمة ليست وهمية ، بل تحتوى على بيانـــات ساعات العمل . ويمكن التجول خلال قاعدة البيانات بتتبع المؤشرات ، فمـــن معرفــة إســم الموظف يمكن معرفة أية مهمة أو كلت إليه وبالتالى أى مشروع يعمل به ، وعدد الســاعات التى قضاها به .



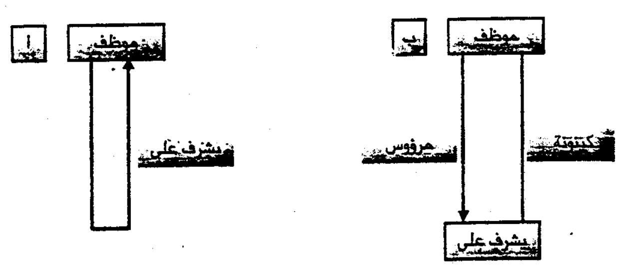
شكل رقم (٦) يوضح الهيكل المنطقى للمؤشرات المرتبطة بالعلاقة موظف/مشروع

ويبين الشكـــل (٧) خاصية أخرى للنموذج الشبكى ، فهو يسمح بأكثر من علاقـــة 1:ن بين سجلين . فالطبيب يرتبط بعدد من المرضى ، إما كمسئول أو كإستشارى .

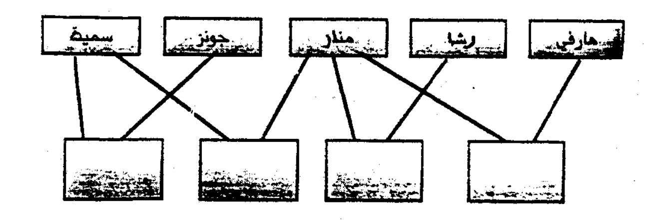


شكل رقم (٧) علاقة مسموح بها فى النموذج الشبكى

وبالاضافة الى منع العلاقة ن:م فى النموذج الشبكى ، لايسمح أيضا بالعلاقة الملتفـــة Involuted relation . فالشخص يمكن أن يكون رئيسا ومرءوسا فى نفس الوقت كما فـــى شكل (٨). وفى النموذج الشبكى لايسمح بتمثيل هذه العلاقة مباشرة ، بل تدخل أيضا كينونـــة تسمى فى مثالنا هذا "المرءوس " ذات سجلات رابطة ، ويصور شـــكل (٨) وقـــائع لـــهذه العلاقة .

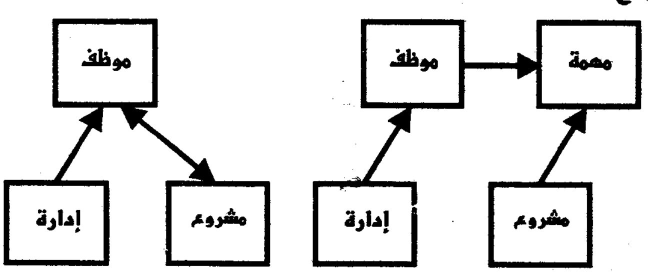


شكل رقم (٨) يوضح معالجة العلاقة الملتفة فى النمـــوذج الشـــبكى أ) تمثيـــل غـــير مسموح به ب) استخدام سجل رابط



شكل رقم (٩) يوضح وقائع لمعالجة العلاقة الملتفة فى النموذج الشبكى

ويمكن إضافة البيانات واسترجاعها فى النموذج الشبكى بطرق مرنة للغاية ، ويعتبر إعادة ضبط المؤشرات عند إضافة أو حذف سجل ما عملية معقدة ، ولكن نظام إدارة القاعدة يتعامل معها آليا . ومع ذلك، فإن كل المنتجات التجارية من تلك النظم، مع كل ماتتيحه من مرونة فى إسترجاع البيانات ، تتطلب أن يكون المستخدم على دراية جيدة بتنظيم المؤشوات لكى يتمكن من التجول فى القاعدة ومعنى ذلك أن الرؤية المقدمة للمستخدم ليست فى الحقيقة منطقية تماما بل تتداخل معها بعض من الرؤية المادية ، وهى نقطة ضعف أساسى فى هذا النموذج .

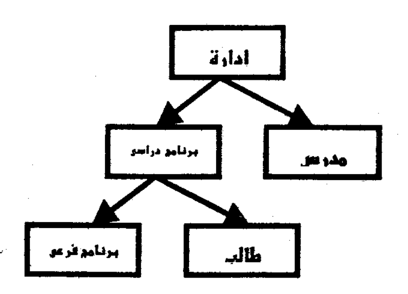


شكل (١٠) التمثيل الشبكى للعلاقة موظف / مشروع / إدارة

(٣) النموذج الهرمى :

كانت قواعد البيانات المبنية على النموذج الهرمى وبرمجيات نظم إدارتها أول ما أنتج من نماذج ومن أكثر النظم شيوعا كان نظام IMS لشركة IBM .

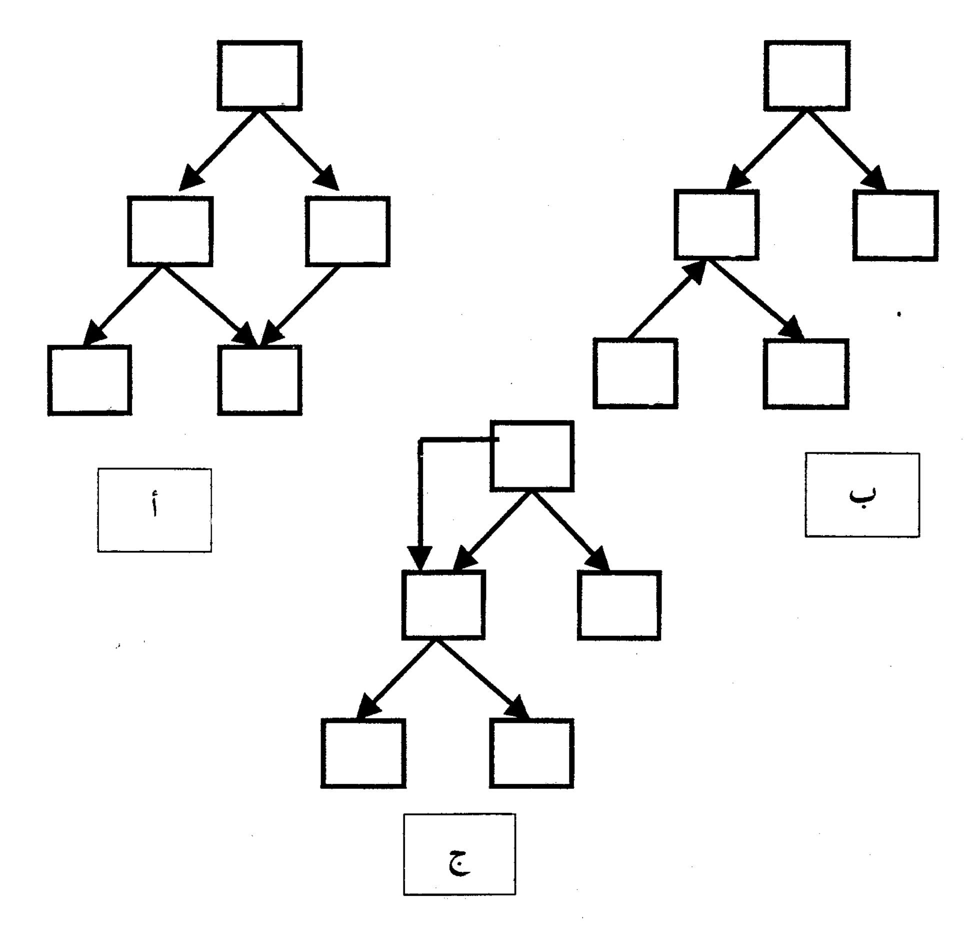
لا تسمح نماذج البيانات الهرمية أيضا بتمثيل العلاقة ن : م ، بل وتفترض قيدا آخـر يميزها هو أن تكون الهياكل فى صورة شجرية . ويبين شكل (١١) تكوينا هرميـــا يشـــكل جزءا من قاعدة بيانات مركزية ويعرف السجل الأعلى ، سجل " الإدارة " بـــالجذر Root ، وتمثل الروابط بين السجلات (وتسمى أحيانا " العقد Nodes ") العلاقات .



شكل (11) هيكل هرمى

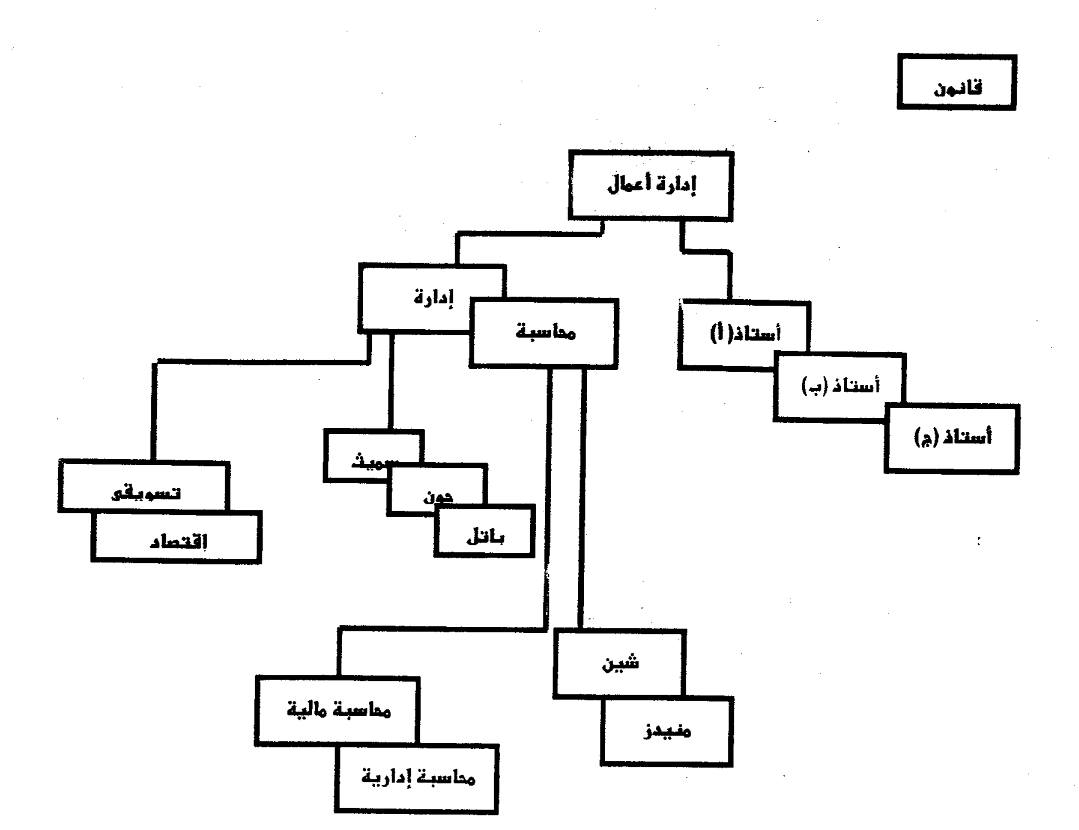
وقد تدير كل إدارة عدة فصول دراسية ، ولكن الفصل الدراسى لا يـــدار إلا بـــإدارة واحدة، وهى علاقة 1: ن كما يتكون كل فصل دراسى من فصول فرعية، ويضم عدة طلبـــة مسجلين به . ويعتبر السجل " الفصل الدراسى " بمثابة " الأب parent " للسجلات التـــالية ، بينما تعتبر السجلات فى المستوى الأدنى " أبناء Children " وتتكون الشجرة من عـــلاقات 1 : ن فقط وتتجه المؤشرات دائما لأسفل .

ويبين شكل (١٢) ثلاثة هياكل غير مسموح بها فى النموذج الهرمى ، ولكن يمكـــن تمثيلها فى النموذج الشبكى . فالهيكل أ ليس شجريا بل شبكيا ، لوجود علاقة بين إثنين مـــن سجلات المستوى الأوسط بسجل واحد من المستوى الأدنى . والهيكل ب به مؤشـــر متجـــه لأعلى، والهيكل يضم أكثر من علاقة بين نوعين من السجلات ، فأيضا ليس هيكلا شجريا .



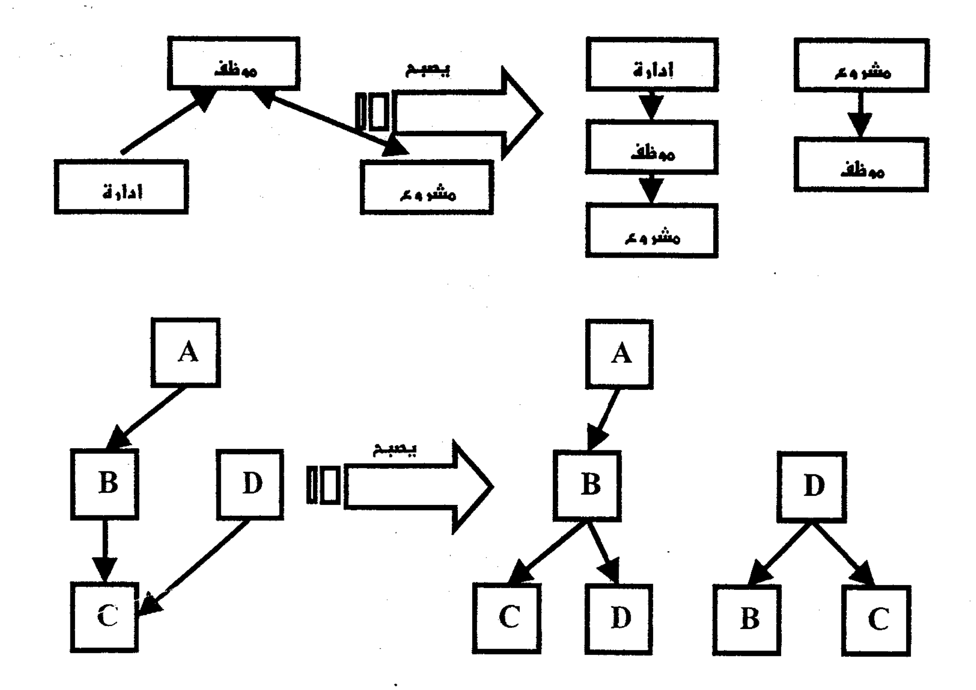
شكل (١٢) يوضح هياكل غير مسموح بها فى النموذج الهرمى

يمثل شكل (١٣) وقائع للهيكل المعطى بشكل (١١). فكل سجل بواقعه بخلاف " إدارة الأعمال " ترتبط بواقعة واحدة فى المستوى الأعلى ، بينما تعتبر الوقائع المرتبطة بنفس السجل الأعلى، مثل فصل " التسويق " والأقتصاد " توائم twins لإرتباطها بنفس الأب . وإذا حذف سجل فى المستوى المتوسط ، فإن كافة السجلات الأبناء له تحذف أيضا فإذا حذف الفصل الدراسى للمحاسبة مثلا ، فمن الطبيعى أن تحذف الفصول التابعة له ، والطلاب المسجلين فيه .



شكل (١٣) وقائع للكينونة "إدارة"

ويتم التعامل مع مشكلة العلاقات ن : م والهيكل الشبكية بإدخال هياكل هرمية بسيطة . ففى شكل (١٤) يتم تمثيل هيكل " موظف / إدارة / مشروع " من خلال تكوينين هرميين بسيطين ، كما يبين الشكل أيضا مثالا آخر ، وتتطلب الهياكل الأكثر تعقيدا لعدد أكثر من الهياكل الشجرية .



شكل (١٤) معالجة هياكل غير هرمية فى النموذج الهرمى

وتتفوق على النماذج الهرمية اليوم نظم قوية مؤسسة على النموذج العلائقى .

٤ - النموذج العلائقى :

يرجع الفضل فى وضع هذا النموذج فى السبعينات إلى E.F.Codd من شركة IBM . وقد تقدم نظم إدارة قواعد البيانات حول النموذج العلائقى مثل RDB/VMS ORACLE ميزات أساسية تتفوق بها على النموذجين السابقين .

يتم فى النموذج العلائقى تخزين البيانات فى جداول . كالجدول الموضح فى الشكل التالى رقم (١٥) وهنا يجب التنبيه فى بعض الإختلافات فى المصطلحات وذلك على الصورة التالية .

الجدول : يطلق عليه المصطلح " Relation " ومن الهام جدا التفرقة بين هذا المصطلح ومصطلح relationship بمعنى علاقة بين شيئين . وكما سنرى لاحقا ، فإنه بينما يمكن تمثيل كافة العلاقات بجداول ، فإن العكس ليس صحيح دائما .

الأعمدة : يطلق عليها " خصائص Attributes " ، فرقم الموظف ، وإسم الموظف ، وراتب الموظف كل هذه " خصائص " فى الجدول المبين . وتتخذ الخصائص قيما فى حدود معينة تسمى " المدى Domain " ، فالمرتب يمكن أن تكون قيمة أرقاما صحيحة أكبر من الصفر .

الصف : يطلق علية إصطلاح Tuple ، ولايمكن أن يتماثل صفان فى نفس الجدول . وتحدد الأعمدة (الخصائص) مستوى العلاقة ، فجدول " الموظف " المبين يمثل علاقة من المستوى الثالث عدد الصفوف : يطلق عليها مصطلح Cardinality .

ومن المفاهيم العلائقية الأساسية " الخصيصة المفتاحية **Key attribute** " وهى أية خصيصة ل يشترك فيها صفان أبدا ، كرقم الموظف . وفى الوصف الكتابى للجدول (شكل ١٥) يوضح خط تحت الخصيصة المفتاحية .

(A) EMPLOYEE

Employee ##	Employee Name	Employee - Salary
134	Smith	12.000
146	Harvey	15.000
139	Jones	4.600
468	Mendez	14.000
201	Patel	9.000

(B) EMPLOYEE (Emp #,Emp_Name , Emp _ Salary)

شكل (15) مثال لجدول (علاقة).

ومن الناحية النظرية الصرفة فإن ترتيب الصفوف ليست لـــه أهميـــة فـــى تحديـــد الجدول ، أمـــا ترتيب الأعمدة فأمر جوهرى ، فإذا تغير ترتيب الأعمدة كنـــا بصـــدد جـــدول جديد . ولكن الكثير من أنظمة إدارة قواعد البيانات التجارية لا تطبق ذلك حرفيـــا ، فيمكـــن التساهل فى ترتيب الأعمدة حيث أن قيمة الحقل فى صف معين تتحدد من إسم العمود الـــذى يوجد تحته بصرف النظر عن ترتيب هذا العمود فى الجدول ، كما أن بعض الأنظمة تســـمح بالتعبير عن ترتيب الصفوف ، كأن تكون مرتبة هجائيا .

وبينما تمثل العلاقات فى النموذجين الشبكى والهرمى بالروابط ، فإنها تمثل فى هـــذا النموذج بالجدول كما يتبين من الشكل (١٦) ويمثل الوصف الكتابى للـــهيكل والمبيـــن فـــى الشكل (١٦) هذا بالإضافة لمدى كل خصيصة فى الجدول – البنيان التصورى للقاعدة .

(I)

EMPLOYEE

Employee #	Employee _ name	Employee _ Salary	Dept _ name
134	Smith	12.000	Sales
146	Harvey	15.000	Sales
139	Mendez	4.600	Production
468	Mendez	14.000	Planning
201	Patel	9.000	Production

DEPARTMENT

DePT _ name	Dept _ location
Sales	Floor 6
Production	Tetherdown
Planning	Floor5

PROJECT

Project _ Name	BUDEGET
Project A	45.000
Project B	500.000
Project D	9.400
Project D	12.000

Asignment

Project _ Name	Employee #	Hours
Project A	146	3.2
Project B	134	9.0
Project A	201	11.0
Project C	146	4.9
Project A	134	6.2
Project B	164	6.1
Project C	201	9.3

(b)

EMPLOYEE (employee # , employee _ name , employee _ salary dept _ name)

DEPARTMENT (dept name , dept _ location)

PROJECT (Project name , budegt) .

ASSIGNMENT (project name , employee # , hours)

شكل (١٦) يوضح

النموذج العلاقى للعلاقة موظف / مشروع / إدارة

أ) العلاقة ، ب) وصف الهيكل

وتلاحظ أن العلاقة ن : م بين المشروعات والموظفين قد مثلت من خلال جدول المهمة ASSIGNMENT " ، والذى يضم الخصيصتين المفتاحيتين فى كل من جدول المشروعات وجدول الموظفين ، ومنه يمكن معرفة أى مشروع يعمل به أى موظف .

ومن مزايا النظم العلائقية أنها توفر للمستخدم بنيايا تصوريا خالصا ، فلا يتعرض لقضايا التخزين أو معرفة المؤشرات أو نظم الفهرسة وخلافة . كما تسمح النظم العلائقية بمعالجة كميات ضخمة من البيانات فى عملية واحدة .

الفصل الثالث

قواعد البيانات العلائقية (الإرتباطية)

الفصل الثالث

قواعد البيانات العلائقية (الإرتباطية)

رغم أن زيادة شهرة قواعد البيانات العلائقية ظهرت بإعتبارهـــا مرتبطـة بثـورة أجهزة الحاسبات الشخصية إلا أن تاريخ ظهور قواعد البيانات العلائقية يعود إلى تاريخ أقـدم بكثير مما يعتقد معظم الناس . ولقد لعب ذلك التاريخ القديم دوراً بالغ الأهمية فـــى تشـكيل قواعد البيانات العلائقية المستخدمة فى الآونة الأخيرة .

ولقد صار هذا الأسلوب ذا أهمية قصـــوى للمنشـــآت المعـــاصرة بسـبب بســـاطة وعموميته .

(١) مفهوم قواعد البيانات المبنيه على العلاقات :

يرتكز مفهوم نظم قواعد البيانات المبنية على العلاقات علـــى أســـاس ملفـــات أو جداول البيانات بدلاً من وحدات البيانات الفردية فى الملف الواحد . وبذلك تتحـدد العمليـات التى تؤدى فى كل الملفات بحيث تكون معالجة البيانات على مستوى أعلى من النظم الأخـرى . وأحياناً تتواجد الطلبات للمعلومات فى ملفات عديدة يمكـن أن تحـدد ببســـاطة وســـهولة ووضوح . وتصبح النظم المبنية على العلاقات سهلة الإستخدام وفـــى الإمكـــان إســـتيعابها وتعلمها بسرعة ، حيث توفر الوقت والجهد والتكلفة . وتصمم هذا النظم على أساس مبنـــى على الرياضيات مما يتيح لها عوائد كثيرة تتعلق بالتكامل والإستيعاب الفورى للبيانات التـــى تتضمنها .

ويبنى مفهوم قاعدة البيانات المبنية على العلاقات على أساس الجداول Tables التى توفر أسلوباً طبيعياً مبسطاً لنقل معلومات فى شكل مختصر . ويشتمل الجدول على عدد مـــن الأعمدة Columns يختص كل عمود منها بخاصية Attribute محددة للأشـــياء أو الـــدلالات التى توصف فى النظام . كما يتضمن الجدول عدداً من الصفوف Rows يشتمل كـــل مدخـــل

فيها على قيمة Value معينة لكل خاصية . وبذلك يمثل الصف الواحد سجلاً Record قائماً بذاته تكون فيه الأعمدة حقولاً Fields للبيانات الخاصة بهذا السجل .

(٢) لغات قواعد البيانات العلائقية:

١- لغة الإستعلام [SQL] Structured Query Language :

تسمى لغة الإستعلام أو لغة البحث وتستخدم غالباً مع برامج مدير قواعد البيانات فى تخليق أو تدمير الملفات كما تستخدم فى البحث الإنتقائى لسجلات دون غيرها بدلاً من إستعراض كل الملفات للوصول إلى السجلات ولغة الإستعلام الإجرائية فهى توصف ماذا نريد WHAT وليس كيف ننجز ما نريد (HOW) حيث يتولى مدير قاعدة البيانات الخطوات الإجرائية دون تدخل من المستخدم فيما يسمى السياحة الآلية فى النظام .

٢- لغة تعريف البيانات DDl :

وتنقسم لغة الإستعلام إلى قسمين ... ، قسم أو لغة تعريف البيانات Data Defeinition Language (DDL) ونعرض إلى بعض أوامرها Create, Derop, Alter .

٣- لغة تحريك وتداول البيانات :

Data Manipulation [DML] Language

وهى القسم الثانى من لغة الإستفسار – الإستعلام – وتستخدم أربعة أوامر لا غير ، ورغم محدودية عددها إلا أنها تحقق كفاءة عالية فى إسترجاع وتحديث وتعديل البيان :

أ- الأمر	إختر	SELECT
	حدث	UPDATE
	أحذف	DELETE
	أحشر	INSERT

وسوف يخصص فصلاً من هذا المؤلف التى تتيح للمستخدم إمكانية التحاور والإستعلام مع قاعدة البيانات المخزنة بالحاسب .

(٣) مميزات قواعد البيانات العلائقية (الإرتباطية) :

مما لاشك فيه أن لغة SQL لم تكن لتحقيق هذا النجاح ما لم تتمكن من بلوغ الأهداف المتوقع إنجازها من قبل الكثيرين . ويعزو النجاح الذى تحقق إزاء إستخدام لغة SQL إلى وجود العديد من العوامل الأساسية ، مثل توفر التقنية الأساسية الخاصة بقواعد البيانات الارتباطية إلى جانب وجود معيار ANSI المستقل ، هذا فضلاً عن كون اللغة ذاتها قوية للغاية وإعتماد معظم قواعد البيانات المستخدمة للغة SQL على بنيات الوحدة التابعة / وحدة الخدمة الحديثة .

تقنية قواعد البيانات الارتباطية :

إن عملية تصميم قاعدة بيانات غير إرتباطية مهمة صعبة بل وإن إستخدامها يعد أمر اصعب . فكما هو الحال فى معظم نظم قواعد البيانات غير الارتباطة ، يعد إنشاء قاعدة بيانات عملية عسيرة نظراً لأنه ينبغى إيضاح البنيات التسلسلية كاملة أو دراسة العلاقات الممكنة كافة قبل إنشاء قاعدة البيانات . فى المقابل تسمح لك قاعدة البيانات الآرتباطية بإمكانية صنع التغييرات بسهولة وبسرعة . كما يمكنك كذلك إضافة وإزالة بنيات قواعد البيانات أثناء تشغيل قاعدة البيانات .

بالإضافة إلى ذلك ، يسمح أيضاً نمط قواعد البيانات الإرتباطية بإمكانية إستنخدام العلاقات التى لم يتم تعريفها مسبقاً . يتيح هذا الأمر للشركات إمكانية تغيير قواعد البيانات الخاصة بها بصورة تدريجية مع مرور الوقت ، الأمر الذى من شأنه أن يقلل من تأثير ومدى أى تغيير .

مزايا وعيوب معايير ANSI :

يعتقد الكثيرون أن معايير ANSI تضمن إمكانية إنتقال البرامج التى تم تحريرها كى ما تستخدم لغة SQL من نظام التشغيل الخاص بقاعدة بيانات إلى

آخر بأقل قدر من المشاكل ، وذلك بنفس الطريقة التى يتم من خلالـــها إنتقـــال برنامج ANISC من جهاز لآخر . يضـــفى هـــذا الأمـــر علـــى العديـــد مـــن المستخدمين شعوراً بأنه إذا كانوا فى حاجة إلى ذلك ، يمكنـــهم التعـــامل مـــع شركة منتجه أخرى لقواعد البيانات . فى الواقع ، ما لم تبذل جـــهوداً خاصـــة لتصميم التطبيق بحيث يتوافق مع معايير ANSI ، فإن الإنتقال من نظام قـــاعدة بيانات إلى أخر سيمثل مشكلة أكبر مما تستحق وذلك بالنسبة للكثيرين .

بيد أن هناك العديد من المميزات لوجود معيار مستقل بخلاف مجرد قابليـــة الإنتقال بين النظم . إذ تتمثل إحدى أكبر المشاكل فى هذه الآونة فى العثور على ذوى الخبرة كى ما يتسنى لهم إمكانية تطوير التطبيقات . ففى الغالب الأعـــم ، يتمتع الأشخاص الذين يتم الإستعانة بهم ببعض المهارات ، إلا أنه مازالت هناك حاجة لتدريبهم فى مهارات أخرى غير أنه رغم أن تفاصيل تركيب SQL قـــد تختلف من شركة منتجه لقاعدة بيانات إلى أخرى ، إلا أن العبـــارات الأساســـية موحدة . ومن ثم يقود ذلك إلى إمكانية تدريب شخص ما فى نظام قاعدة بيانـــات جديدة من خلال التركيز على نقاط الإختلاف بين SQL الخاصة بالشركة ولغـــة SQL القياسية فقط وذلك بدلاً من بدء لغة جديدة تماماً . يعنى ذلك أن الإنتقـــال من نظام قواعد بيانات إلى آخر سيستغرق جهوداً أقل مما لو كان الأمـــر قائمـــاً دون وجود لغة قياسية مستقلة .

لغـة إستعلامات عاليـة الكفـاءة :

قبل ظهور لغة SQL إلى حيز الوجود ، إســـتعانت معظـــم نظـــم قواعـــد البيانات بأسلوب معين للوصول إلى قاعدة بيانات بصورة تفاعلية وأسلوب آخـــر للوصول المبرمج . أجبر هذا الأمر المبرمجين على تعلم لغـــات متعـــددة حتـــى يتسنى لهم الوصول إلى قاعدة البيانات الخاصة بهم . بالإضافة إلى ذلـــك ، لـــم تكن لغة الإستعلامات التفاعلية عالية الكفاءة للدرجة التى تكفـــى لحـــل أنـــواع معينة من المشاكل .

فضلاً عن ذلك هناك لغـــة ثالثة كان يتم الإستعانة بـــها لتعريـــف البنيـــة الخاصة بقاعدة البيانات . وقد جعل ذلك الأمر أكثر صعوبة مما ينبغى أن يكـــون

بالنسبة للعديد من المبرمجين . بيد أن وجود لغة SQL قد أدى إلى حـل هـذه المشكلة . إذ أنـها تقدم لغة إستعلامات واحدة فقط للقيام بـالأغراض الثلاثـة كافة . من ثم ، يمكن إستخدام نفس اللغة بصورة تفاعلية وبصـورة مبرمجـة فضلاً عن إمكانية الإستعانة بها لتعريف البنيات الخاصة بقواعد البيانات .

ي بنية الوحـدة التابعـة / وحدة الخدمـة :

هناك ميزة أخرى لدى إستخدام نظم قواعد البيانات التى تعتمد علـى لغـة SQL بشكل أساسى ألا وهى تطور معظم نظم قواعد البيانات من خـلال بنيـة الوحدة التابعة / وحدة الخدمة ، مما يجعله أمراً طبيعياً أن تتناسب مع الشـبكات الخاصة بأجهزة الكمبيوتر الشخصية الحديثة . وقد أدى ذلك أيضاً إلى سـهولة قياس إمكانيات الكمبيوتر الذى يتم تشغيله على وحدة الخدمة الخاصـة بقـاعدة البيانات ، الأمر الذى يتيح فى المقابل إمكانية إنفاق أموال اقـل علـى أجـهزة الكمبيوتر .

(٤) مكونات قاعدة البيانات الإرتباطية :

كى يتسنى لنا إستيعاب كيفية الوصول إلى قاعدة بيانات إرتباطية فإننا بحاجة إلـى معرفة المكونات الأساسية لقاعدة البيانات الإرتباطية وكيفية تناسقها معاً .

١/٤ جداول وصفوف البيانات :

تحتفظ قاعدة البيانات بالبيانات داخل سلسلة تتألف مـن واحـد أو اكـثر مـن الجداول . ويشبه كل جدول فى هيئته الجدول الإلكترونى ، إذ يتم تنظيم البيانات داخل سلسلة من الصفوف والأعمدة . فيتوافق كل صف مع أحد السجلات فى ملـف كمـا يمثل نسخة مميزة من مجموعة من البيانات . على الجانب الآخر ، يتوافق كل عمـود مع أحد الحقول فى ملف ويمثل سمة واحدة من مجموعة من البيانات .

تصور جدولاً يحتوى على المعلومات الخاصة بأحد العملاء . حيث يحتوى كـل صف فى الجدول على معلومات خاصة بعميل واحد ، بينما يضم كل عمود فى الجدول سمة مميزة عن هذا العميل كإسمة أو عنوانه .

في الغالب الأعم ، يطلق على البيانات الموجودة في العمود بيانات كلية . مما يعنى أن الأوامر الموجودة فى العمود لا يمكن تقسيمها . على سبيل المثال ، تصور حقلاً إسمه Name يتكون من ثلاثة أجزاء فرعية First name و Middle name و last name . ومن ثم ، لا يمكن إعتبار حقل Name حقل كلى نظراً لأنه توجد إمكانية لتقسيمه إلى أجزاء أصغر. على الجانب الآخر ، يصبح الحقل الذى يحتوى على كمية غير مجزئة خاصة بإسم حقلاً كلياً ، وذلك نظراً لعدم وجود تقسيمات داخل الحقل .

إن المجموعة المتكررة ما هى إلا حقلاً يحتوى على أكثر من نسخة واحدة لقيمة البيانات . من ثم يعتبر أى حقل يمثل مصفوفة أو مجموعة بمثابة مجموعة متكررة ونظراً لأن المجموعة المتكررة كذلك غير كلية لذا لا يمكن تمثيلها من خلال عمود واحد فقط .

أما فيما يتعلق بنوع البيانات فهو مرتبط بكل عمود ، الأمر الذى يتم من خلاله تعريف نوع البيانات الذى تنوى تخزينه فى العمود . وتنقسم أنواع البيانات بوجه عام إلى أربعة أنواع :

- الأرقام – مثل الأعداد الصحيحة والأعداد ذات العلامة العشرية الثابتة Desimal Floating Point Numbers .
- سلاسل الأحرف – دائماً ما تشغل سلاسل الأحرف ذات الطول الثابت نفس المساحة من الجدول بينما تقوم سلاسل الأحرف ذات الطول المتغير بتخزين الرموز الموجودة فى سلسلة الأحرف فحسب ، فضلاً عن بعض المعلومات التى تتيح لقاعدة البيانات إمكانية متابعة طول سلسلة الأحرف .
- قيم التاريخ / الزمن – يحتوى هذا النوع من البيانات على قيمة التاريخ أو قيمة الزمن أو مركب من قيمة التاريخ والزمن . وسوف تقوم كل وحدة خدمة خاصة بقاعدة البيانات بتحديد معنى تلك القيم بالضبط إلى جانب كيفية تخزينها .

• قيم ثنائية – وهو حقل غير منسق تقوم بتخزينة بنفسك بصفة أساسية . إذ يمكنك تخزين عناصر مثل الصور والقطات الصوتية ، بل والبرامج التطبيقية كذلك بإستخدام هذا النوع من البيانات الثنائية .

إلى جانب تخزين قيمة بيانات في عمود ، هناك علامة خاصة متاحة أيضاً تعرف بإسم Null . وتشير هذه العلامة إلى ما إذا كان العمود يحتوى على قيمة صالحة للإستخدام أم لا ، فإذا كانت هناك قيمة صالحة للإستخدام لدى العمود ، فسيتم ضبط العلامة على False ويعرف العمود فى هذه الحالة بإسم Not Null . أما إذا لم تقم بتعيين قيمة هذا العمود ، فسيتم ضبط هذه العلامة عندئذ على True وسيعرف العمود فى هذه الحالة بإسم Null . ملحوظة : لا تعنى القيم Null أنها Empty : ينبغى ألا يحدث لبس بين سلسلة أحرف Empty و Null. حيث تمثل سلسلة الأحرف Empty قيمة ، بيد أن Null تعنى أنه لا توجد ثمة لا توجد ثمة قيمة مرتبطة بهذا العمود .

٢/٤ الفهارس Keys :

فى نمط قاعدة البيانات الإرتباطية ، لا يوجد ثمة تنظيم بشأن كيفية تخزين الصفوف فى الجدول من الناحية النظرية . بيد أنه من الجانب العملى ، أنت بحاجة إلى آلية تسمح لك بإمكانية تحديد صف أو مجموعة من الصفوف بصورة سريعة . تعرف هذه الآلية بإسم الفهرس .

يعتبر الفهرس بمثابة بنية خاصة من قواعد البيانات تحتفظ بمجموعة من قيم الحقول التى تم تخزينها على نحو يقلل من وقت البحث إلى أقصى حد ممكن . ويتم تنفيذ الفهرس بمثابة قائمة معكوسة بحيث يشبه القائمة المستخدمة فى نمط قواعد البيانات المفهرسة . حيث تسمح هذه القائمة لوحدة الخدمة الخاصة بقاعدة البيانات بالبحث سريعاً خلال قائمة من القيم الموجودة فى الفهرس بغرض تحديد الصفوف المطلوبة من الجدول . ويتم تخزين الفهارس بصورة مستقلة عن الجدول ومن ثم يمكن إضافتها وإزالتها بصورة سريعة .

من الجدير بالإعتبار أن العمود يعتمد على حقل آخر حينما يوجد قيمة واحدة ممكنة فقط فى الحقل الثانى متعلقة بقيمة معينة خاصة بالعمود الأول . إذا كان ذلك

يتسبب فى حدوث بعض اللبس بالنسبة إليك ، فعليك بتصور ما يلى : يوجد لديك جدول يحتوى على عمود الإسم وعمود رقم التأمين الإجتماعى . نظراً لأن كل رقـــم تـــأمين إجتماعى سيضحى مميزاً لكل فرد علاوة على أنه يوجد لدى كل فرد إسم واحد فحسـب فإن حقل الإسم متوقف على رقم التأمين الإجتماعى . لا حظ أن العكس ليس صحيحـاً ، بإعتبار أنه من الممكن أن يوجد لدى الإسم المعنى أكثر من رقم تـــأمين إجتمـــاعى . بصورة أخرى ، ربما يكون هناك أكثر من شخص يحمل إســـم John Smith فـــى البلاد ، من ثم كل شخص سيكون لدية رقم التأمين الإجتماعى الخاص به .

مـــن ناحية أخرى ، يطلق على قائمة الأعمدة الموجودة فى الفـــهرس مصطلـــح Key . إذ ينبغى أن يكون هناك Key لدى كل جدول بحيث تكون القيمة الخاصـــة بـــه مميزة لكل صـــف موجود فى هذا الجدول . ويعرف ذلك بإسم Primary Key . علـــى سبيل المثال ، قد يوجد لدى الجدول الذى يحتوى على معلومات بشأن العملاء عمـــود يطلق عليه Customer Id يعتبر بمثابة Primary Key بالإضافة إلى ذلك ، يعرف الـــ Key الذى يحتوى على أكثر من عمود أيضاً بإسم Composite Key .

أمـــا Keys الأخرى الموجودة فى الجدول فيشار إليها بمثابـــة Secondary Keys . وعلى العكس من Primary Keys ، فإن Secondary Keys ليســت بحاجة لأن تكون مميزة . إنما هى تساعدك فى تحديد مجموعة فرعيـــة شـــائعة مـــن الصفوف فى الجدول فحسب .

وعندما تتوافق مجموعة من الأعمدة فى جدول واحد مع تعريـــف Primary Key فى جدول آخر ، يطلق على هذه المجموعـــة مـــن الأعمـــدة Foreign Key وتعتـــبر Foreign keys مفيدة عندما ترغب فى إنشاء علاقة بين جدولين . على سبيل المثال ، فى جدول العملاء قد يمثل كود المنطقة أحد الأعمدة . وبالتالى يمكنك إضافة الجـــدول الخاص بكود ZIP إلى قاعدة البيانات ، هذا الجدول الذى يحتوى على قائمة من أرقـــام كود ZIP ، كل مع إسم المدينة المتوافقة معه . من ثم يمكنك تحديد إسم المدينة محـــل إقامة العميل حال تحديد أى كود ZIP . بهذه الطريقة ، يمكنـــك مـــلء إســـم المدينـــة أتوماتيكياً عندما يقوم العميل بإدخال كود ZIP الخاص بها .

بالإضافة إلى ذلك يمكنك إختيار صنع علاقة - علاقة أساسية - تحول من خلالها وحدة الخدمة الخاصة بقاعدة البيانات دون إتاحة إمكانية إصحة صف إلى الجدول ما لم تتواجد القيمة الموجودة فى الأعمدة الخاصة بـ Foreign Key بمثابة Primary Key فى الجدول المتوافق . من ثم ، إذا إخترت إنشاء العلاقة المذكورة أعلاه ، فيمكنك إدراج صف داخل جدول العميل فقط فى حالة ما إذا كان كود ZIP الخاص بالعميل يوجد فى الجدول الخاص بكود ZIP.

فضلاً عن ذلك ، تتسم الفهارس بميزة أخرى ألا وهى إمكانية إستخدام فهرس كى ما تتأكد أن القيمة الموجودة فى الجدول قيمة مميزة . فرغم أن الـ Primary Key لجدول لا بد أن يكون مميزاً ، إلا أنه فى بعض الأحوال قد تود أن يكون الـ Secondary Key مميزاً ايضاً .

٣/٤ طرق العرض :

تمثل طريقة العرض أحد أهم المفاهيم الموجودة فى قواعد البيانات الإرتباطية . وتعد طريقة العرض ببساطة بمثابة " جدول إفتراضى " تم إنشاؤه من خلال واحد أو أكثر من الجداول الأساسية الموجودة فى قاعدة البيانات . وتنقسم طرق العرض إلى قسمين : طرق عرض قابلة للتحديث وطرق عرض غير قابلة للتحديث . وكما يتضح من الإسم ، فإن طرق العرض القابلة للتحديث هى طرق عرض يمكن تحديثها كما أنها عادة ما يتم إنشاؤها من خلال جدول واحد . أما طرق العرض غير القابلة للتحديث فيتم إنشاؤها بوجه عام من خلال إثنين أو أكثر من الجداول أو أنها تحتوى على أعمدة يتم حساب القيم الخاصة بها بنفس الأسلوب .

٤/٤ التنسيق :

يعتبر التنسيق بمثابة وسيلة لتصنيف البنية الخاصة بقاعدة البيانات . فمن الناحية النظرية ، كلما كانت قاعدة البيانات أكثر تنسيقاً كلما كان ذلك أفضل . وفيما يلى المستويات الأربعة الرئيسية من التنسيق الموجودة بصورة شائعة فى تصميمات قواعد البيانات العادية :

- **البنية غير المنسقة** : لا يتم فرض ثمة قواعد على بنية قاعدة البيانات .

* **النموذج العادى الأول** : ينبغى كل حقل أن يكون كلياً . ولا يجوز إستخدام مجموعات مكررة والحقول المركبة .
* **النموذج العادى الثانى** : ينبغى أن يعتمد كل حقل غير أساسى على الـ Primary Key بأكملة . فلابد ألا يعتمد الحقل على جزء من الـ Composite Primary Key الأساسى المركب فقط . كما أنه ينبغى كذلك أن توجد قاعدة البيانات فى النموذج العادى الأول .
* **النموذج العادى الثالث** : لايمكن أن يعتمد حقل غير أساسى على حقل آخر غير أساسى . كما أنه لابد أن توجد قاعدة البيانات كذلك فى النموذج العادى الثانى .

بالإضافة إلى ذلك ، توضح أيضاً نظرية قواعد البيانات الإرتباطية النموذج العادى الرابع والنموذج العادى الخامس إلى جانب النموذج العادى Boyce / Codd . وينتج عن النماذج الأخيرة تلك قاعدة بيانات يوجد لديها العديد من الجدول كى ما تقدم أداء جيداً .

من ناحية أخرى ، يتعذر عليك بناء قاعدة بيانات إرتباطية غير منسقة نظراً لأنه غير مسموح بالمجموعات المتكررة والحقول المركبة . (بالطبع هناك طرق تتجاوز أيضاً عن هذه القيود فى معظم وحدات الخدمة الخاصة بقواعد البيانات) . وتتمثل أهم الأشياء التى ينبغى إستيعابها أثناء الإنتقال من مرحلة التنسيق فى عملية إنقسام الجدول الضخمة الخاصة بقواعد البيانات إلى جداول أصغر وأصغر . ويتم تنفيذ ذلك تحت مسمى تقليل تكرار البيانات .

على سبيل المثال ، فى قاعدة البيانات المنسقة بصورة فعلية ، لن تقوم بتخزين المعلومات الخاصة بالمدينة والولاية مع عنوان الشخص ، على إعتبار أنه يمكنك الحصول على هذه المعلومات من خلال كود ZIP الخاص بالشخص . سوف تقوم بعد ذلك بإضافة جدول آخر إلى قاعدة البيانات يستخدم كود ZIP بمثابة Primary Key ، ومن ثم سيكون لديك معلومات المدينة والولاية المتوافقة مع كود ZIP . غير أنه للحصول على عنوان الشخص ، ينبغى الآن الدخول على جدولين مختلفين ، الأمر الذى

يستغرق وقتاً أكبر من الدخول على جدول واحد به جميع المعلومات الخاصـــة بعنــوان العميل .

بناءاً على ما تقدم وحتى يتضح للقارئ مفهوم العلاقات التى تقــوم عليــها قواعــد البيانات فى أحد أشكالها – يمكن لنا أن نوضح من خلال الجدول التالى مفهوم العلاقات فـــى بيئة قواعد البيانات كالتالى :

رقم البائع	إسم البائع	منطقة البيع	كمية المبيعات
١.	أحمد محمد عبد الله	القاهرة	١٠,٠٠٠
٢.	إبراهيم عدلى	الزقازيق	١٥,٠٠٠
٣.	على حسن	بنها	٥,٠٠٠
٤.	محمود إبراهيم	الأسكندرية	٣٠,٠٠٠
٥.	حارس عطيه	أسيوط	١,٠٠٠
٦.	مجدى عزيز	أسوان	١٠,٠٠٠

جدول رقم (١) يوضح قاعدة البيانات المبنية على العلاقات للبائعين

يتضمن الجدول السابق هيكلاً واضحاً للبيانات متكون من أربعة أعمدة رئيسية يعــبر كل منها على دلالة موضوعية Heading معينة مثل رقم البائع وإســم البــائع ومنطقــة البيع وكمية المبيعات . وتشكل هذه الدلالات الموضوعية على جزء محدد وأساسى مــن الجدول التى بدونها يصبح هذا الجدول عديم الجدوى . وفى نفس الوقــت يوجــد لــهذا الجدول ستة مداخل Entries يخصص لكل منها صفاً معيناً أما إذا كان عدد البائعين أكـثر من ستة فيخصص لهذا الجدول عدد أكبر من الصفوف يساوى عدد البائعين .

ويعتبر نموذج الجدول أو المصفوفة فى نظم قواعد البيانات شائعاً جــداً ، ومنتشــر الإستخدام فى نظم معالجة البيانات بسبب بساطتها وسهولة تفهمها .

ويعالج الجدول الذى يشبه التركيب كعلاقة فى حد ذاته وخاصة عندما يساعد فى الوصول إلى أوضاع معينة . وتطبق عليه نظرية العلاقات الرياضية مما يؤدى إلى تصميم مدخل موحد وثابت يطلق عليه المدخل المبنى على العلاقات .

ولا يتصف نظام قاعدة البيانات المبنى على العلاقات بأنه النظام الذى يتداول الجداول فقط . فقاعدة البيانات التى تتداول وتعالج الجداول ليست بالضرورة قاعدة بيانات مبنية على العلاقات كما يدعى كثير من الموردين . أما مجموعة القواعد الرئيسية ذات الفائدة العملية فى مجال التكامل والأستعادة المستمدة من النظرية الرياضية فيمكن أن يعبر عنها بما يلى :

- إشتمال الجدول على نوعية واحدة من السجل الذى يشتمل على عدد ثابت من الحقول تعنون بوضوح . وفى العادة تتضمن قاعدة البيانات على جداول عديدة مختلفة حيث تحفظ فيها السجلات المختلفة .
- يجب أن تتميز الحقول بوضوح فى إطار الجدول وألا تتكرر .
- السجلات تكون فريدة وغير مكررة أيضاً .
- ترتيب السجلات فى الجدول يكون غير محدد Indeteminate حيث ترد فى أى ترتيب كما لا يوجد تتابع محدد لها من قبل .
- تستمد قيمةValue الحقول فى أى عمود من المجال Domain الخاص بقيم بالحقل الذى يستخدم لأنواع كثيرة ومختلفة للحقول فى جداول عديدة .
- تنتج الجداول الجديدة على أساس مقارنة قيم الحقل مع نفس المجال فى جدولين متواجدين . ويعتبر إعداد وتشكيل الجداول الجديدة من الجداول المتواجدة جوهر المعالجة المبنية على العلاقات .

ونتيجة معالجة قاعدة البيانات التى تشتمل على مجموعة من الجداول يمكن إخراج جداول جديدة فيها . ويلاحظ أن العمليات فى إطار قاعدة البيانات تكون على أساس جداول كاملة بدلاً من سجلات فردية فى الجداول . أى أنها ذات خاصية معالجة أقوى من خصائص المعالجة التى تتمثل فى " سجل بواسطة سجل Record – By – Recerd " للنظم الأخرى .

وتعتبر القاعدة الخاصة بالحقول والسجلات فى الجداول ضرورية . ويجب أن تكـون الجداول مستطيلة الشكل تتكون مع عدد متساوى من العناصر Elements فى إطار كل صـف كما لا يسمح فيها بالمجموعات المكررة ولكن قد يتغلب على ذلك بواسطة إستخدام سـجلات أو صفوف أكثر بإستخدام جداول أكثر .

أما فكرة المجال الخاصة بالقيم المحتملة فهى ذات أهمية قصـوى لمدخـل قواعـد البيانات المبنية على العلاقات ، حيث تشير كل قيم العمود إلى خاصيـة Attribute واحـدة للعناصر التى تتمثل بواسطة الصفوف فى الجدول . فقد يعبر أحد الأعمدة كما فـى المثـال السابق على رقم البائع ، والآخر على إسم البائع والثالث على منطقة البيع .. إلخ . وحتـى يمكننا تحقيق التماسك أو الترابط يجب أن تستمد كل القيم فى العمود الواحد من مجال القيـم الممكنة .

أما المجال فيعبر عنه بمجموع القيم الممكنة المتصلة بالخاصية المعينة . ولايحتـاج الجدول الواحد إلى تواجد كل هذه القيم الممكنة . وعلى سبيل المثال يمكن أن يعـرف إسـم الموظف برقمة أو الكود الخاص به الذى قد يشتمل على أحد الحـروف الهجائيـة المكونـة للحرف الأول من إسمه متبوعاً بثلاثة أرقام كما فى المثال التالى : أ/ ١٢٣ ، م/٤٥٠ وبذلـك فإن مجال رقم الموظف يتضمن الحرف الهجائى والرقم ١ ، والرقم ٢ ثم الرقـم ٣ وبذلـك يتوفر للنظام ما يقـرب من ٢٦٠٠٠ رقم . ويستخدم رقم الموظف فى المجال الذى عـرف به . كما أن الرقم الذى لا يوجد فى المجال كأن يستخدم حرفين هجائيين ورقمين فحسـب لا يعتبر رقماً صحيحاً .

وبذلك تعتمد المعالجة المبنية على العلاقات على القيم فى إطار العمود فى الجـدول وكيفية ترابطها مع قيم أخرى فى موقع آخر .

وتقوم قواعد البيانات بثلاث عمليات أساسية تبنى على العلاقات هى :

١ - الإختيار : Selection

تعتبر هذه العملية أبسط العمليات لقاعدة البيانات ، حيث يختار من الجدول الأصلـى صفوفاً Rows معينة . ويعالج الجدول الأصلى بعدئذ لإنتاج جداول جديـدة . وتسـتخدم معايير إختيار محددة تقوم بالتقاط صفوفاً محددة من الجدول الأصلى . وبذلـك يتكـون

الجدول الجديد الذى يختار فيه كل الصفوف التى تقابل متطلبات معايير الإختيار المحددة. ويشتمل هذا الجدول على بعض أو كل الصفوف من الجدول الأصلى . وتعتمد نتيجة الجدول على قيم الحقول التى تتضمن فى معايير الإختيار .

فمثلاً من الجدول السابق عرضه رقم(١) الخاص بقاعدة البيانات المبنيه على العلاقات يمكننا إخراج جدول جديد يتصل بكمية المبيعات التى تكون ٥ الآف أو اقل كما هو موضح فى الجدول الجديد التالى :

كمية المبيعات	منطقة البيع	إسم البائع	رقم البائع
٥٠٠٠	بنها	على حسين	٠٣
١٠٠٠	أسيوط	حارس عطية	٠٥

جدول رقم (٢) جدول جديد مستمد من الجدول الأصلى لقاعدة البيانات الخاصة بالبائعين .

٢- الوصل أو الربط : Join

قد يحتاج الإنسان فى الغالب إلى جمع المعلومات من عدة جداول مختلفة . وبذلك فهناك حاجة إلى وصل أو ربط هذه الجداول المختلفة لإنشاء جدول جديد منها . وهناك عدة أساليب تستخدم للوصل أو الربط التى منها أسلوب الوصل الطبيعى الأكثر إنتشارا . وقد ينجز الوصل عن طريق مضاهاة Matching القيم التى تحدث فى جدولين أو اكثر ، حيث تشتمل هذه الجداول على خاصية مشتركة بينها كرقم الموظف ، أو رقم العميل أو رقم البائع .. إلخ . وتنتج عملية الوصل جدولاً جديداً يشتمل على سجلات مستمدة من جداول أصلية . وفى مثال الجدول الخاص بالبائعين رقم (١) ووصلة بجدول آخر عن قاعدة بيانات العملاء كما فى الجدول التالى :

منطقة البيع	إسم العميل	رقم العميل
القاهرة	حامد عبد اللطيف	٠٠,٠١
أسوان	وليد محمد	٠٠,٠٢
الأسكندرية	عمرو خالد	٠٠,٠٣
الزقازيق	محمد داوود	٠٠,٠٤
بنها	كمال الدين حسين	٠٠,٠٥
الأسكندرية	نعيم أبو العزم	٠٠,٠٦
القاهرة	يوسف صبرى	٠٠,٠٧
الأسكندرية	فوزى معاذ	٠٠,٠٨

جدول رقم(٣) قاعدة بيانات مبنية على العلاقات للعملاء

وعند وصل هذا الجدول بالجدول الخاص بالبائعين رقم (١) المتسع فـإن الخاصيـة التى تربط الجدولين معاً هى منطقة البيع أى أنها المجال المشترك بين الجدولين وبذلـك يمكننا أن ننتج جدول جديد لمنطقة الأسكندرية مستمد من الجدولين الأصليين الخـاصيين بالبائعين والعملاء .

إسم البائع	رقم البائع	إسم العميل	رقم العميل
محمود إبراهيم	٠٤	عمرو خالد	٠٠٠٣
سعيد نور الله	٠٧	نعيم أبو العزم	٠٠٠٦
فتح الله على	٠٨	فوزى معاذ	٠٠٠٨

جدول رقم (٤) قاعدة بيانات لجدول جديد عن منطقة المبيعات " الأسكندرية "

٣- الأسقاط : Projection

فى هذه العملية يختار عمود أو أعمدة Columns معينة من الجدول الأصلى لإنتاج جدول جديد يحتاج إليه للإجابة على إستفسارات معينة . وبذلك يشتمل الجدول الجديد على أعمدة معينة تختار لذاتها بينما يهمل عرض باقى الأعمدة . أى أن عملية الإسقاط تختص بالأعمدة بينما تختص عملية الإختيار بالصفوف . والجدول الجديد الناتج من ذلك قد يشتمل على بعض السجلات أو الصفوف المكررة ، علماً بأن السجلات المكررة تعتبر ظاهرة غير مسموح بها فى الجداول المبنية على العلاقات وبذلك فإن عملية الإسقاط تشتمل على أعمدة أقل مما يتواجد فى الجدول الأصلى .

وقد تستخدم عملية الإسقاط فى إطار جدول البائعين رقم (١) لإنتاج جدول جديد عن معلومات كمية المبيعات وبذلك يمكن أن ينشأ هذا الجدول مشتملاً على عمودين فقط خاصين بكمية المبيعات . وإسم البائع كما فى الجدول التالى :

كمية المبيعات	إسم البائع
١٠,٠٠٠	أحمد محمد عبد الله
١٥,٠٠٠	إبراهيم عدلى
٥,٠٠٠	على حسن
٣٠,٠٠٠	محمود إبراهيم
١٠,٠٠٠	حارس عطية
١٠,٠٠٠	مجدى عزيز

جدول رقم (٥) قاعدة البيانات

ويشتمل هذا الجدول الجديد على ستة سجلات تحتلان عمودين فقط يختاران من الإسقاط على الجدول الأصلى لقاعدة البيانات عن البائعين . وفى العادة يمكن إسقاط الأعمدة فى أى نظام أو ترتيب فى الجدول الجديد . وقد يكون الإسقاط بغية الحصول على كمية المبيعات أو لأى هدف يمثله العمود فى الجدول الأصلى لإنتاج جدول جديد .

وفى هذه الحالة يكون الجدول الجديد مكوناً من خمسة صفوف أو سجلات فقط بدلاً من ستة .

حيث أن كمية المبيعات أو القيمة ١٠,٠٠٠ المكررة مرتين فى الجدول الأصلى تظهر مرة واحدة فى الجدول الجديد الناتج من الإسقاط .

(٥) نشأة لغات الإستعلامات كأساس لقواعد البيانات العلائقية (الإرتباطية) :

من الجدير بالإعتبار أنه دون وجود لغة – Structured Query Language (SQL) لما ذاع صيت قواعد البيانات الإرتباطية كما هو الحال اليوم . إذ تعزو أهمية قواعد البيانات الإرتباطية بصفة رئيسية إلى كيفية تطور صناعتها فضلاً عن المعايير التى أسهمت فى هذا التطور . ومما لاشك فيه أن مميزات قواعد البيانات الإرتباطية كذلك قد أضفت المزيد من الأهمية المؤثرة على ذلك النوع من قواعد البيانات .

ويرجع نشأة لغات الإستعلام إلى المقال الذى نشره دكتور **Codd** فى يونيو عام ١٩٧٠ بعنوان :

Arelationl Model of Data for large Shared Data Banks

Communications of the Association فى صحيفة

For Computing Machinery

وقدم هذا المقال نظرية رياضية عن كيفية تخزين ومعالجة البيانات بإستخدام أبنية جدولية للبيانات . وتعتبر هذه النظرية بمثابة الأساس الذى بنيت عليه قواعد البيانات العلائقية الحديثة .

١ – لغة SEQUEL ومشروع System / R

لقد كانت فكرة إنشاء قاعدة بيانات إرتباطية أمراً ملحاً إزاء وجود العديد من الأسباب ، وقد إعتمدت شركة IBM مشروع بحث فى منتصف السبعينات يهدف إلى إنشاء نظام قواعد بيانات يعتمد على تلك النظرية السالف ذكرها . وقد أصبح هذا المشروع معروفاً بإسم Systems / R . بالإضافة إلى العمل فى قاعدة البيانات نفسها ، كان العمل مستمراً أيضاً لإجراء تطوير على لغة الإستعلامات الخاصة بقواعد البيانات .

وقد طور الباحثون العديد من اللغات المختلفة ، بما فـي ذلـك لغـة معروفـة بإسـم SEQUEL(Structured English Query Language) .

فى أواخر السبعينات ، قررت شركة IBM إصدار مشـروع Systems /R لبعـض العملاء الرئيسيين بغرض تقييمة . وكجزء من عملية التطوير ، تم تغيـير إسـم لغـة الإستعلامات إلى SQL ، رغم أنه إستمر نطقـها SEQUEL . وقـد أوشـك مشـروع System /R على إنتهاء فى عام 1997 ، عندمـا قـررت شـركة IBM أن النظريـات الخاصة بقواعد البيانات الإرتباطية التى تم تطويرها عن طريق Codd منذ عشر سنوات جاهزة آنذاك كى ما يتم إحالتها إلى منتج تجارى .

٢- نظام قاعدة بيانات Ingres ونظام قاعدة بيانات Oracle:

لم تكن شركة IBM هى الشركة الوحيدة التى تعمل فـي مجـال قواعـد البيانـات الإرتباطية خلال فترة السبعنيات . فقد قام بعض الأساتذة فى معامل Berkeley بجامعـة جنوب كالفورنيا بإنشاء قاعدة بيانات إرتباطية معروفة بإسم Ingres . وقد كـانت لغـة الإستعلامات الخاصة بقاعدة البيانات تلك معروفـة بإسـم QUEL ، وهـى إختصـار لمصطلح Query Language . وكما هو الحال مع لغة SQL ، فقد إعتمدت هذه اللغـة كذلك على نموذج متطور التركيب من اللغة الإنجليزية .

ولقد تم تطوير Ingres كذلك على نظام تشغيل Unix ، وذلـك علـى العكـس مـن مشروع Systems /R والذى يتم تشغيله على نظـام التشـغيل الخـاص بأجـهزة IBM الرئيسية . وقد كان يتم تشغيل نظم Unix آنذاك بوجة عـام علـى أجـهزة الكمبيوتـر الصغيرة مثل Digital's VAX ، والتى كانت لاتتطلب تكاليف عادية لإقتنائها وتشـغيلها كما هو الحال مع الأجهزة الرئيسية . وقد أدى ذلك إلى ذيوع شهرة قاعدة البيانات فـي العديد من الجامعات التى إسعانت بها لتعليم المفاهيم الأساسية للتطبيقات الخاصة بقواعد البيانات الإرتبارطية .

فى النهاية ، ترك بعض الأستاذة معامل كمبيوتر Berkeley فى عام 1980 وقـاموا بتأسيس شركة معروفة بإسم Relational Technology ,.Inc لإنشاء إصدار تجارى مـن قاعدة بيانات Ingres . ومازالت Ingres توجد حتى الآن بمثابة قاعدة بيانات تجاريـة ،

إلا انها لم تتمكن من تحقيق شهرة بعض تطبيقات قواعد البيانات الأخرى بالإضافة إلى ذلك ، هناك شركة أخرى يطلق عليها Relational Software ., Inc تم تأليفها فى أواخر السبعينات لإنشاء نظام قواعد بيانات إرتباطية خاص بها . وقد أطلق على نظام قواعد البيانات الخاص بها Oracle ، والذى ظهر لأول مرة إلى حيز الوجود فى عام 1997 ، مما جعله أول نظام قاعدة بيانات إرتباطية متاح تجارياً . ومثلما هو الحال مع مشروع System/R الخاص بشركة IBM ، يعتمد نظام قواعد بيانات – Oracle على لغة SQL ، غير أنه يتم تشغيله على أجهزة الكمبيوتر الصغيرة VAX مثل Ingres . وقد أضحى Oracle فى الآونة الأخيرة المدعم الرئيسى لنظم قواعد البيانات الإرتباطية على مستوى العالم أجمع .

٣- الإصدارات الخاصة بشركة IBM :

فى عام 1981 ، أصدرت شركة IBM أول نظام تجارى لقواعد البيانات الإرتباطية بالنسبة لها ، وكان يطلق عليه SQL/Data System (SQL/DS) . وقد كان SQL / Ds متاحاً لنظام تشغيل Virtual Machine (VM) وقد تم إنشاؤه خصيصاً من أجل مراكز الكمبيوتر التى ترغب فى تقديم وسيلة لمعالجة الإستعلامات العشوائية الخاصة بقواعد البيانات الصغيرة نسبياً .

ويعزو إنتهاج شركة IBM لهذا الأسلوب إلى سببين . أولهما يتمثل فى عدم رغبتها فى القضاء على المبيعات التى كانت تتحقق إزاء وجود نظام قواعد البيانات التسلسلية الناجح بصورة عالية (IMS) Information Management Systems والذى تم تشغيله على الأجهزة الرئيسية الضخمة التى تعتمد بصورة أساسية على نظم تشغيل (MVS) Multiple Virtual Storage . أما السبب الثانى فيتمثل فى كون نظام SQL/DS يستغرق وقتاً أكبر بكثير من IMS مهما كان قدر عمل . ومن خلال حصر تقنية قواعد البيانات الإرتباطية على إمكانيات محدودة ، تمكنت شركة IBM من ممارسة الخبرة العملية مع هذه التقنية إلى جانب إستمرار التعامل مع العملاء فى ذات الوقت .

فى عام 1983 ، أصدرت شركة IBM أول نظام قاعدة بيانات إرتباطية لتشغيله على نظام تشغيل MVS يطلق عليه Data Base/2 ، والذى يعرف بإسم نظام قاعدة بيانات DB2 بصورة أكثر شيوعاً . وقد إشتهر DB2 فى المرحلة الأولى بأنه ليس عالى الكفاءة

نظراً لكونه بطئ للغاية من حيث الأداء . وقد كان هذا الأمر حقيقيا بالمقارنـــة بقـاعدة بيانات IMS ، فقد كانت IMS تتسم بالتناسيق والفعالية على نحو أكبر من DB2 أنـــذاك من ثم ، كان رد فعل شركة IBM يتمثل فى إقتراح مفاده أنـــه ينبغـى إسـتخدام IMS لمعالجة العمليات ذات الحجم الكبير .

فضلاً عن ذلك ، كانت شركة IBM تصف DB2 كذلك بمثابة منتج إستراتيجى . وفى تصريح لشركة IBM ، يعنى ذلك المصطلح أن العميل الذى تنصب دائرة تركـيزه علـى المستقبل ينبغى أن يقوم بتشغيل نظام قاعدة بيانات DB2 فى الحال . رغم أن ذلك الأمـر ساهم فى زيادة مبيعات DB2 ، إلا أن الوضع لم يكن كذلك بالنسبة للأداء . بيد أنه مـع مرور الوقت ، ساعد عاملان فى تغيير اداء DB2 بصورة مؤثرة . يتمثل العـامل الأول فى وقوع DB2 تحت تأثير العديد من التغييرات الرئيسية للإصدار ، حيث كان كل تغيـير يقود إلى تطوير مهم فى الأداء . فى الوقت نفسه ، إنخفضت أسعار الأجـهزة بصـورة هائلة ، الأمر الذى يعنى أن العميل قد يشترى أجهزة كمبيوتر ذات إمكانيات أكبر بكثـير وبنفس قدر السعر . ومن الجدير بالإعتبار أنه لا يوجد أمر يتمكن من حل مشـكلة الأداء كما يؤدى وجود المزيد من الأجهزة .

ملاحظة : " إن إستخدام ذاكرة كبيرة يعتبر أمراً جيداً إلا أن إستخدام ذاكرة أكبر أمـر أفضل : عند التعامل مع قاعدة البيانات الإرتباطية ، لا يوجد ثمة شئ يؤدى إلى تحسـين الأداء كإضافة الذاكرة . إذ أن إضافة المزيد من RAM إلى الكمبيوتر يسـمح لقـاعدة البيانات بإمكانية تقليل مقدار المدخلات والمخرجات المطلوبه لمعالجة الإستعلام . فضـلاً عن ذلك أن إضافة RAM إلى الذاكرة المؤقتة الخاصة بالقرص سـيؤدى أيضـاً إلـى تحسين أداء العمل ".

لغة SQL القياسية :

فى عام 1982 ، بدأ المعهد الوطنى الأمريكى للقياسات المعياريـة (ANSI) عمليـة تحديد معايير بالنسبة للغة الإستعلامات الخاصة بقواعد البيانات الإرتباطية . وتم عقب ذلـك إستعراض العديد من اللغات المختلفة على مدى السنوات الأربع التالية ، بما فى ذلـك SQL و QUEL ؛ إلا أنه نتيجة لدعم شركة IBM لنظام قواعد بيانات DB٢ ، تم تحديد لغة SQL فى النهاية كى ما تصير اللغة القياسية . لكن على الرغم من إختيار لغة SQL بمثابة اللغـة

القياسية ، إلا ان هناك بعض الإختلافات الرئيسية بين التركيب الفعلى المستخدم من لغة SQL من خـــلال DB2 والتركيب النهائى الذى تم إعتماده من قبل معهد ANIS فى عام 1986 .

وإزاء وجود العديد من الثغرات لدى التركيب النهائى الذى تم التصديق عليه فى عام **1986** ، قام معهد ANIS بمراجعة المعايير للمرة الثانية فى عام 1989 ، ثم للمرة الثالثة فى عـــام 1992 . وقـد أضحت هذه المعايير معروفة بإسم SQL – 86 و SQL – 89, SQL – 92 . ونظــراً لأن تلـك المعايير سمحت بمستويات متعددة من التوافق ، زعمت العديد من الشركات المنتجــة لقواعـد البيانــات أن قواعد البيانات الخاصة بها تدعم معايير SQL الخاصة بمعهد ANSI .

فى النهاية خضع **Ingres** للضغط الواقع عليه ، وقام فى عام **1986** بإضافة الدعم الخـــاص بلغة SQL فضلاً عن لغة QUEL . منذ ذلك الحين ، قامت جميع الشركات المنتجة لقواعـد البيانـــات تقريباً بإضافة الدعم الخاص بلغة SQL إلى قواعد البيانات الخاصة بها . بل أن هناك بعض من الشـركات المنتجة لقواعد البيانات التى كانت تتعامل مع نظم قواعد البيانات غير الإرتباطيـــة قـــاموا عندئــذ بتدعيــم **Structured Query Language** أيضاً .

من ناحية أخرى ، إتبعت الشركات الأخرى المنتجة لقواعد البيانات الإرتباطية قيـــادة **Oracle** وإنحازوا إلى جانب لغة SQL ، وذلك مثل **Sybase** و **Digital Informix** و **Microsoft** . وفى معظم الأحيان ، كانت الشركات المنتجة تتجاوز عن الأجهزة الرئيسية وتوجه جهودها نحـــو أجـــهزة الكمبيوتر الصغيرة و / أو وحدات الخدمة الخاصة بأجهزة الكمبيوتر الشخصية المتصلة بالشبكة ، الأمر الذى لايلبث وإن يمثل قراراً صائباً فى ضوء إنخفاض معدل الطلب على قواعد البيانات الخاصة بأجهزة الكمبيوتـو الرئيسية ، بينما – من ناحية أخرى – يظل معدل الطلب على الأجهزة الرئيسية ذاتها ثابتاً .

ملحوظة : الأجهزة الرئيسية ووحدات خدمة **Unix** : نظراً لأننى تعاملت مع العديد من أجـــهزة الكمبيوتر ذات الأحجام المختلفة على مدى سنوات متعددة ، فإننى تعلمت أنه لايوجـــد فرق كبير بين أجهزة الكمبيوتر الرئيسية وبعض من أضخم وحـــدات خدمـــة **Unix** المتاحة فى الأسواق ، وذلك كما يعتقد الكثيرون . فلقد تـــم تطويـــر معظـــم التقنيـــة المستخدمة فى وحدات خدمة **Unix** من أجل أجهزة الكمبيوتر الرئيسية فى المقـــام الأول ، كما تم من ناحية أخرى دمج مناهج التصميم المبتكرة التى تم الإستعانة بـــها لإنشاء وحدات الخدمة المتاحة من **Unix** داخل أجهزة الكمبيوتر الرئيسية . ويعنـــى هذا المزج بين التقنية والتصميم أن أجهزة الكمبيوتر الرئيسية ستظل متاحـــة علـــى المدى الطويل بمثابة بديل لوحدات خدمة **Unix**.

الفصل الرابع

منهجية إستخدام التصميم المنطقى فى قواعد البيانات العلائقية

الفصل الرابع
منهجية إستخدام التصميم المنطقى فى قواعد البيانات العلائقية

مقدمة:

إن نمذجة بيانات النظام المرتقب أمر حيوى فتوافر البيانات أمر ضرورى ، ما نعنيه بالتوافر ليس زيادة الكم بل الكم المناسب من البيانات وبالقدر اللازم والضرورى فأى تزايد فى كم البيانات ليس ميزة ذاتية بل عبئ يكلف جهد ومال دونما طائل خلفة ويرجع هذا التزايد الكمى فى الكثير من الأحيان إلى أن بعض المنشئات تأخذ بمبدأ مركزية إدارة المعلومات أو إستخدام قاعدة بيانات موحده ، ودون أن يشعر أحد تبدأ الإدارات والأقسام فى إنشاء أرشيفات صغيرة لخدمة أغراضها ، وتدريجياً تتضخم الأرشيفات الوليده وتتحول فى غيبة من الناس إلى صور مكررة من الأرشيف المركزى مسببة مشاكل أبرزها :

أ- تكرارية البيانات دونما حاجة فعلية إليها .

ب- عدم تكاملية البيانات .

ج- عدم إتاحة فرض قواعد تأمين موحدة للبيانات .

د- إستهلاك أصول رأسمالية إضافة إلى جهد الأفراد .

و- تحديث أى بيان لا يعنى بالضرورة تحديثه باقى مخازن البيانات ، مثل تلك المشاكل تحتم نمذجة بيانات النظام وإعادة ترتيبها ووضعها فى أبسط صورة NORMALIZATION بما يدرء كل المخاطر والمثالب السابقة .

والسؤال الآن ، وما الفرق بين مخازن البيانات فى خرائط التدفق وبين النماذج المقترحة للبيانات ؟ والإجابة أن مخازن البيانات فى خرائط التدفق تشير إلى بيانات ساكنة نسبياً مقارنة بالمعالجات التى تحدث أو تضيف أو تحذف أو تحولها إلى بيانات أخرى أو معلومات ، أما نمذجة البيانات هنا فإنها تركز على

نموذج أو نماذج أفضل لهياكل البيانـــات بصـــرف النظـــر عـــن المعالجـــات ممـــا يســـاعد على إنشاء نظام معلومات جيـــد دون مشـــاكل .

إن البيانات فى أى نظام معلومـــــات هـــى إجمـــالى الخصـــائص المتكـــررة حـــول كيانـــات النظـــام ســـواء كـــانت كيانـــات داخليـــة أو خارجيـــة ، أو كيانـــات فيزيائيـــة أو منطقية مثل السلع – العملاء – المـــوردون – الموظفـــون أو تعـــبر عـــن أمـــر توريـــد ، فاتورة دفع ، عقود ، أو تعبر عن أمـــاكن ، ولا شـــك أنـــه توجـــد بيـــن الكيانـــات ترابـــط وعلاقات RELATIONS وقد ترقى هذه العلاقات إلـــى حـــد أنـــها فـــى ذاتـــها تولـــد بيانات ، خذ مثلا العلاقـــة بيـــن الكتـــاب والمســـتعير فـــهى علاقـــة يعـــبر عنـــها بكلمـــة يستعير ، والعلاقة بين المستهلك والسلعة هـــى يشـــترى وعلـــى نفـــس النـــهج الموظـــف والمكان ، والموظف والبطاقة العائلية وعلى نفـــس المنـــوال فـــإن للنظـــم الفرعيـــة فـــى النظام الأكبر علاقات تبادلية قد تولد بيانـــــات جديـــدة .

وفى هذا الفصل ســـوف يتـــم إلقـــاء الضـــوء علـــى الطريقـــة المتبعـــة لتصميـــم قواعد البيانات العلائقيـــة وكيفيـــة تطبيـــق وإســـتخدام بعـــد الأدوات الأساســـية لتصميـــم قواعد البيانات كالأدوات الخاصـــة كلغـــة SQL ، وخرائـــط الكيانـــات والعلاقـــات ERD وقاموس البيانات ... إلخ وســـــوف نتعـــرض فيمـــا يلـــى المنـــهج المتبـــع فـــى عمليـــة التصميـــم .

(١) خطوات عملية التصميم :

ربما يكون تصميم قـــاعدة بيانـــات إرتباطيـــة ســـهلا أو صعبـــا ، وذلـــك حســـبما تختار أن يكون وبوجة عام ، سيتم الإستعانة بمنـــــهج مكـــون مـــن خمـــس مراحـــل كمـــا يلى :

١- تحديد المشـــكلة

٢- طرح وتسجيل الأفكـــــار .

٣-تشكيل الكيانات والعلاقـــــات .

٤-بناء قاعدة البيانـــــات .

٥-إنشاء التطبيـــق .

سوف يتم مناقشة المراحل الأربع الأولى من خلال الفصل الحالى أما المرحلة الخامسة فستكون موضع إهتمام باقى أجزاء الكتاب .

١- تحديد المشكلة

رغم أنه قد يبدو تحديد المشكلة أمرا سهلا ، إلا أن الأمر أصعب بكثير مما يبدو فينبغى أن يمثل عرض المشكلة إستيعابا للهدف الذى تحاول المؤسسة تحقيقه فى ذات الوقت الذى تحاول فيه التأكيد على أهم إحتياجات العمل الأساسية . فإذا كنت ترغب فى إستبدال تطبيق موجود يمكنك إستخدام ذلك التطبيق بمثابة قاعدة أساسية لحل المشكلة . أما إذا كنت تقوم بإنشاء تطبيق لأول مرة فإنه أمر غاية فى الأهمية أن تفهم ماهية المشكلة التى تحاول حلها أولا . فبصفة عامة فإن لك أن تدرك أنك قمت بحل المشكلة بالفعل إذا لم تكن قادرا على تحديدها فى المقام الأول ؟!

ولتوضيح كيفية تحديد المشكلة نرى بعض الأمثلة : فإذا قام ...... الأفراد فى Amazon . Com بتحديد مشكلتهم ، فقد تبدو كالآتى : نحن نرغب فى تأسيس Amazon . Com بمثابة علامة تجارية من خلال بيع الكتب للمستهلكين عبر الإنترنت بأقل تكلفة للمستهلك وأفضل خدمة متاحة ، فضلا عن إنشاء حصة من السوق تضمن إستقرارية وضع الشركة على المدى الطويل . من ناحية أخرى ، قد ترغب شركة متخصصة فى إرسال طلبيات الكتالوجات عبر البريد الإلكترونى فى حل هذه المشكلة : نحن نرغب فى تحسين الخدمات المتعلقة بكيفية إستقبال طلبات العملاء عبر التليفون لتقليل الأخطاء التى يمكن وقوعها وتطوير الخدمة للعملاء . أما مورد الإلكترونيات ذو النشاط المحدود قد يعلن أن مشكلته تتمثل فى الآتى نحن فى حاجة إلى تطوير عملية الرقابة على الجرد لزيادة عدد مرات إجراء عملية الجرد فى كل عام والإحالة دون وجود بضائع مهملة فى المخزون الإحتياطى . وفى هذا الكتاب ، سوف نتناول قاعدة بيانات شائعة من أجل الأمثلة المقترحة كافة . بالإضافة إلى ذلك سوف يتم كذلك تحديث تطبيق سبق كتابته منذ عدة سنوات مضت يطلق عليه Car Collector والذى من شأنه أن يتابع مجموعة السيارات اللعبة .

ولا شك أن تناول مثل هذا التطبيق سوف يوضح مدى أهمية شبكة الإنترنت بالنسبة لمن توجد لديهم هواية الجمع فقد جعلت مواقع الويب مثل Ebay هذا الأمر أسهل بكثير لهؤلاء الذين يرغبون في بيع وشراء المجموعات التي قام الهواة بجمعها ، .

وحينما يتم أخذ هذا التطبيق كمثال يمكن تحديد المشكلة كالآتى : إننى أرغب فى تحديث تطبيق Car Collector كى ما يتضمن الدعم للأنواع الأخرى من الألعاب والمجموعات التى قام الهواة بتجميعها ، هذا فضلا عن إضافة الدعم لممارسة تجارة الألعاب مع من يوجد لديهم هواية الجمع عبر الإنترنت . إلى جانب السمات الجديدة التى سوف أقوم بإضافتها ، سوف أقوم كذلك بتغيير إسم التطبيق إلى Toy Collector .

ملحوظة : لم يكن الغرض من تطبيق Toy Collector المستخدم كمثال فى هذا الكتاب أن يكون تطبيقا متكاملا ، وإنما هو نظام يوضح التقنيات المختلفة لبناء تطبيقات قواعد بيانات فى Visual Basic . من ثم ، قد لايكون تصميم قاعدة البيانات متكاملا كالتطبيق المعد بغرض تجارى ، كما أنه لن توجد كافة السمات المتاحة من خلال التطبيق المعد بغرض تجارى فى تطبيق Toy Collector . بل أن بعض تراكيب البيانات فضلا عن السمات المستخدمة الموجودة فى التطبيق - قد تبدو مبالغ فيها ، رغم أنها ضرورية لتصوير مختلف النقاط المتعددة . لذا فعليك بالتركيز على التقنيات التى سوف يتم عرضها ومحاولة إستيعاب السبب وراء القيام بها وكيفية عملها بدلا من التركيز على السبب وراء إضافة هذه السمة أو عدم ذكر معلومات بعينها .

٢- طرح وتسجيل الأفكار :

بمجرد إدراك المشكلة التى يهدف التطبيق إلى حلها على نحو متكامل نكون بحاجة إلى تصميم قاعدة بيانات لإعداد المعلومات المرتبطة بالتطبيق . وتتمثل الخطوة الأولى فى هذه العملية فى إلقاء نظرة على المشكلة ومحاولة تحديد

المعلومات والوظائف التى قد تكون هناك حاجة إليـــهما لحـل مثـل هـذه المشـكلة . ويطلق على الخطوة عملية طرح وتســجيل الأفكــار .

إن طرح وتسجيل الأفكار ماهى إلا عملية مناقشـــة وتسـجيل الأفكـار المقترحـة بغض النظر عن إمكانيـــة تحقيقــها . وتسـاعدك هـذه المســألة فـى تحديـد كافـة المعلومات التى أنت بحاجة للإحتفاظ بها فضـــلا عــن جميــع المــهام المطلــوب مــن التطبيق القيام بــها .

من الأفضل عقد جلسات طرح وتسجيل الأفكـــار بحيــث يحضرهــا كــل شــخص مشترك بالمشروع فى حجرة واحدة . فإن ذلــك سيســاعد علــى وجــود عــدد كبــير من الحاضرين بقدر الإمكـــان . بـل أنـــه ينبغــى أن يحضــر الجلســة كـل شــخص مشترك فى هذه العمليـــة بــدء مــن المســتخدمين النــهائيين ومــرورا بـالمبرمجين وحتــى الإدارة .

بالإضافة إلى ذلك ينبغى سرد كـل مســألة تــم طرحــها بحيـث تكـون واضحــة لجميع الحاضرين ، حتى إذا كانت هذه المسألة مشـــابهة لفكــرة تــم كتابتــها بــالفعل . وعقب الإجتماع ، ينبغى تنظيــم وجمــع الأفكــار المتشــابهة معــا بمثابـة عنصــر واحد . ثم يجب بعدئذ تصنيف الأفكار ، إمــا كمــهام ينبغــى للتطبيــق القيــام بــها أو كمعلومات هناك حاجة للإحتفــاظ بــها .

من ناحية أخرى ، مـــن المـــهم أن نــدرك أن جلســة طــرح وتســجيل الأفكــار يمكن أن تسفر عن وجود بعض الأفكار التى قد تتســم بكونــها غــير عمليــة . غــير أنه فى هـــذه المرحلــة مــن العمليــة ، لاينبغــى أن يســاورك القلــق إزاء الناحيــة العملية . حيث أن الأمــر الأهــم بصــورة أكــبر أن تكــون متكاملــة . ففــى بعــض الأحيان ، قد تثبت الأشياء التى تبدو غــير عمليــة فــى هــذه المرحلــة أنــها ســهلة التنفيذ لاحقا ، بينما قد يتكشف فيما بعد أن الأفكـــار التــى تبــدو ســهلة التنفيــذ فــى المرحلة الأولى لاتستحق الوقت والجهد اللذين يتـــم بذلــهما لإنجازهــا .

فضلا عن ذلك ، من المهم كذلك ألا نسخر من أيـــة أفكــار ، بغــض النظــر عــن كونها تبدو سيئة . يبدو هذا صحيحا ، لاسيما فيمـــا يتعلــق بالأفكــار المقترحــة مــن

قبل الحاضرين ذوى الخبرة الأقل . ففى الغالب الأعم ، قد تقود أفكارهم وتعليقاتهم إلى إستيعاب أفضل فيما يتعلق بكيفية عمل التطبيق .

أ- طرح وتسجيل الأفكار الخاصة بتطبيق Toy Collector :

فمن خلال المشاركة فى جلسة طرح وتسجيل الأفكار نجد أن تلك الجلسة قد تسفر عن القائمة التالية من الوظائف المطلوب القيام بها من خلال تطبيق Toy Collector :

- متابعة السلع المتوفرة حاليا فى المجموعة .
- إنشاء تقارير عن السلع المتاحة فى المجوعة .
- تحديد موقع إحدى اللعب فى المخزن .
- الإحتفاظ بالقائمة البريدية الخاصة بالعملاء الحاليين والرئيسيين منهم .
- إنشاء صفحة ويب تحتوى على قائمة بالعناصر المتاحة حاليا للبيع .
- إنشاء صفحة ويب تحتوى على قائمة بالعناصر المطلوبة .
- إنشاء قائمة HTML بالنسبة لموقع ويب Ebay .
- تقييم حالة اللعبة ..
- متابعة عمليات الشراء والبيع .
- إصدار طلبية لبيع لعبة أو أكثر إلى أحد العملاء .
- إصدار طلبية شراء لعبة من أحد العملاء .

وسوف يتم الإستعانة بهذه المهام بغرض الإحتفاظ بقاعدة البيانات بحيث تحتوى على عناصر البيانات التالية :

- إسم اللعبة
- المنتج
- الوصف
- عام إنتاج اللعبة
- تاريخ الطلبية
- تاريخ الإستلام
- تاريخ الشحن
- رقم بطاقة الإئتمان

- القيمة
- السعر الخالص سداده
- السعر المطلوب
- الحالة
- معايير تقييم اللعبة
- نوع اللعبة
- المعلومات الخاصة
- بالطلبية
- تكاليف الشحن
- ضريبة المبيعات
- تاريخ الإنتهاء
- إسم العميل
- الإسم الأول للعميل
- الإسم الأخير للعميل
- الحرف الأول للإسم الأوسط الخاص بالعميل .
- محل إقامة العميل
- مدينة العميل
- ولاية العميل
- صورة المجموعة .

ب-إستعراض النتائج :

عقب عقد جلسة طرح وتسجيل الأفكار نكون بحاجة لاستعراض المعلومات التى تم جمعها ومحاولة التخلص من المعلومات المكررة . وفى الغالب الأعم ، قد ندرك أثناء إجراء هذا الإستعراض أن هناك معلومات أخرى متعلقة بالموضوع قد تتسم بكونها مفيدة للغاية ، ومن ثم ينبغى إضافتها إلى هذه القائمة .

ج - تعيين النتائج بالنسبة لأنواع البيانات :

تتمثل الخطوة الأخيرة فى جلسة طرح وتسجيل الأفكار فى تعيين عناصر البيانات لسلسلة من أنواع البيانات . إذا أنه عقب إستعراض المعلومات الخاصة بجلسة طرح وتسجيل الأفكار ، يتم تجميع قائمة من عناصر البيانات التى أسفرت عنها الجلسة ، فضلا عن نوع بيانات Visual Basic ووصف موجز للعناصر ، وذلك كما هو موضح فى الجدول التالى رقم (١)

عناصر البيانات

عنصر البيانات	نوع البيانات	الوصف
Customer Name	String	إسم العميل
Street	String	الشارع الذى يقيم فيه العميل
City	String	المدينة التى يقيم فيها العميل
State	String	الولاية التى يقيم فيها العميل
Zip	String	كود ZIP المتوافق مع محل إقامة العميل
Phone	String	رقم تليفون العميل
EmailAddress	String	عنوان البريد الإلكترونى الخاص بالعميل
MailingList	Boolean	صحيحا ، إذا كان العميل يرغب فى إستقبال إخطارات بصورة دورية
Order Number	Long	رقم يعمل على تعريف الطلبية بصورة مميزة
OrderStatus	String	الوضع الحالى للطلبية
Dataordered	Data	تاريخ إصدار الطلبية
DataShipped	Data	تاريخ إرسال الطلبية
ShippingCost	Currency	تكاليف شحن الطلبية
SalesTax	Currency	مقدار ضريبة المبيعات
CreditCardNumber	String	رقم بطاقة الإئتمان المستخدمة لشراء اللعبة
ExpirationDate	String	تاريخ إنتهاء بطاقة الائتمان
Inventoryld	Int	أداة تعريف مميزة للعنصر الموجود فى المجموعة
ToyName	String	إسم اللعبة التى تم تجميعها
Manufacturer	String	منتج اللعبة
ToyType	String	نوع اللعبة
ToyDescription	String	وصف اللعبة

عنصر البيانات	نوع البيانات	الوصف
MintValue	Currency	قيمة اللعبة إذا كانت فى حالتها الأصلية السليمة
Condition	Long	وصف رقمى لحالة اللعبة
Question	String	سؤال يستخدم لتقييم حالة اللعبة
TrueValue	Currency	القيمة الحقيقية للعبة وفقا لحالتها
DatePurchased	Date	تاريخ شراء اللعبة
Image	Picture	صورة اللعبة

ملحوظة :عند إنشاء تطبيق فى مجال التجارة الإلكترونية ، فإن كيفية التعامل مع بعض الأمور التى يتم تداولها على المستوى العالمى تعد أحد أهم الأشياء التى نكون فى حاجة للتخطيط إليها . وفى معظم الأحيان ، تتمثل الصورة الوحيدة التى تؤثر بها هذه الأمور على تصميم قاعدة بيانات فى الحاجة إلى ضم معلومات إضافية مثل Country و CurrencyType، كما أنك سوف تحتاج كذلك لأن تسمح بوجود حيز إضافى للحقول الأخرى مثل كود ZIP ، إذا لا يمثل عدم وضع هذه الحقول ضمن قاعدة Toy Collector سببا كافيا كى لا تضم هذه الحقول فى التطبيق الخاص بك .

٣- تشكيل الكيانات والعلاقات :

تتمثل الخطوة القادمة فى عملية تصميم قاعدة البيانات الإرتباطية فى ترجمة المعلومات التى تم الحصول عليها من خلال جلسة طرح وتسجيل الأفكار داخل تصميم قاعدة البيانات .

أ- عملية تشكيل الكيانات / العلاقات :

تعتبر عملية تشكيل الكيان / العلاقة (والتى تعرف أيضا بإسم عملية تشكيل (Entity Relationship (E/R) بمثابة وسيلة لوصف العلاقة بين الكيانات

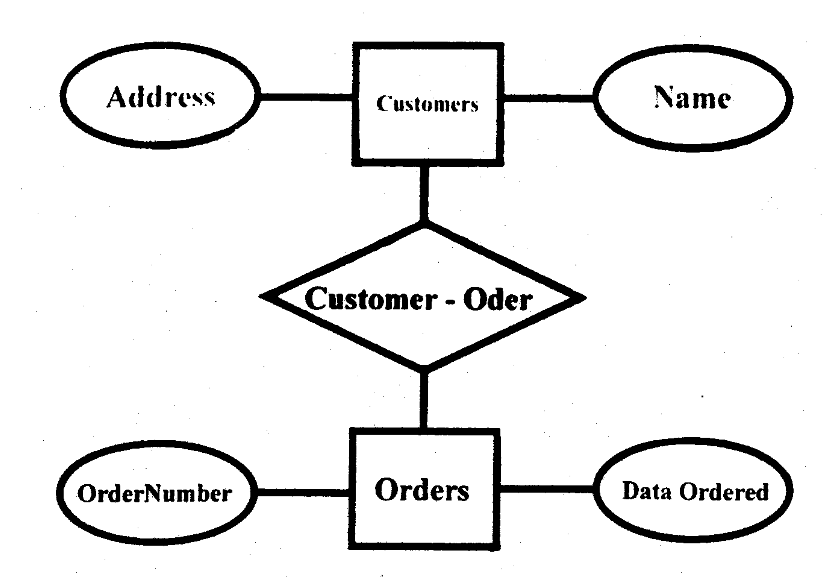
وذلك لصياغة العلاقات فى أبسـط صورهـا لتكـون عونـا ومـددا علـى تصميـم قاعدة بيانات النظـام .

أى أن ER هى توثيق لبيانات المنشأة بطريقـة تلخيصيـة عـن طريـق تعريـف أنواع كينونات البيانات وخصائص كل منها والعلاقـات المتداخلـة بينـها .

والكينونة : هى شيئ ما قادر على التواجد بصـورة مسـتقلة وتـود المنشـأة بالإحتفاظ ببيانات عنه . يمكن أن يكـون نـوع الكينونـة (١) عنصـرا بيئيـا أو (٢) موردا أو (٣) عمليـة جاريـة لـها أهميـة للمنشـأة بحيـث يجـب أن يوثـق مـع البيانـات.

ويرتبط بالكيان مجموعة مـن السـمات (الخصـائص) ممـا يسـاعد ذلـك علـى وصف الكيان . إذ يوجد لدى كل عميـل فـى مثالنـا إسـم وعنـوان . كمـا يوجـد لدى كل لعبة إسم ومنتج وبطريقة مماثلـة يوجـد لـدى كـل طلبيـه رقـم الطلبيـة وتاريخ إصدارها ويتم تشكيل العلاقات بين كـائنين كمـا هـو الحـال مـع العمـلاء والطلبيات حيث يقوم العميل بإصدار طلبية بخصـوص إحـدى اللعـب .

عند رسم نمط E/R ، فإن يتـم إسـتخدام المسـتطيلات للتعبـير عـن الكيانـات والقطع الناقص للتعبير عـن السـمات والشـكل المنحـرف الـذى يشـبه المعيـن للتعبير عن العلاقات وفى شـكل (١) ، يمكـن مشـاهدة نمـط E/R بسـيط يوجـد لدية كيانان (Customers أو Orders) وبحيث يكـون لـدى كـل كيـان خـاصيتين (Address و Name بالنسـبة لكيـان – Customers) ، (Order Number و Data Ordered بالنسبة لكيـان Orders) فضـلا عـن وجـود علاقـة واحـدة ألا وهـى (Customer – Order) .



الشكل (١) كيفية تصميم قاعدة بيانات بسيطة بإستخدام نمط E/R

ب – تعريف الكيانات والسمات (الخصائص)

تتمثل الخطوة الأولى في هذه العملية في إستعراض قائمة من عناصر البيانات الموجودة في جدول (١) والبحث عن المجموعات الشائعة . وبينما تقوم بتصفح قائمة عناصر البيانات ، تظهر ثلاث مجموعات رئيسية على الفور : المعلومات الخاصة بالعميل والمعلومات الخاصة بالمخزون والمعلومات الخاصة بالطلبيات . إذ يمثل كل من هذه المجموعات كيان رئيسي في قاعدة بيانات Toy Collector .

في نفس الوقت نكون بحاجة إلى معرفة الكيانات المختلفة والسمات الخاصة بها من خلال التطبيق . فقد تجد بعض السمات الأخرى التي تتميز بسهولة إضافتها إلى قاعدة البيانات ، بل وأنها ستزيد من قيمة التطبيق من وجهة نظر المستخدم .

**** المعلومات الخاصة بالعميل :**

يحتوي جدول (٢) على قائمة من عناصر البيانات المتعلقة بأحد العملاء ، فضلا عن بعض العناصر الأخرى التي طرحت أثناء تجميع القائمة . ويعتبر

العثور على بعض عناصر البيانات الإضافية في هذه المرحلة أمرا طبيعيا للغاية ، على إعتبار أنه أصبح لدينا إستيعاب أفضل لمكونات قاعدة البيانات . ولقد قمت في هذه الحالة بإضافة حقول لتعريف الزمن الذى تم عنده إضافة العميل إلى قاعدة البيانات (Date Added) والمرة الأخيرة التى تم فيها تحديث المعلومات . كما قمت كذلك بإضافة حقل يطلق عليه Comments يسمح للمستخدم بإمكانية تسجيل أى تعليقات قد تكون لدية عن عميل معين .

جدول (٢) المعلومات الخاصة بالعملاء

إسم العمود	نوع البيانات	نوع VB	الوصف
Customerld	Int	Long	أداة تعريف مميزة للعميل
Name	Varchar(64)	String	إسم العميل
Street	Varchar(64)	String	الشارع محل إقامة العميل
City	Varchar(64)	String	إسم المدينة التى يقيم فيها العميل
State	Char(2)	String	إسم الولاية التى يقيم فيها العميل
Zip	Int	Long	كود ZIP الخاص بعنوان العميل
Phone	Varchar(32)	String	رقم تليفون العميل
EmailAddress	Varchar (128)	String	عنوان البريد الإلكترونى الخاص بالعميل
DateAdded	Datetime	Data	تاريخ إضافة العميل إلى قاعدة البيانات
DateUpdated	Datetime	Data	تاريخ آخر تحديث تم إجراؤه للمعلومات الخاصة بالعميل
MailingList	Bit	Boolean	عندما يكون صحيحا ، فإن ذلك يعنى أن العميل يرغب فى إستقبال رسائل بصفة دورية .
Comments	Varchar (256)	String	تعليمات خاصة با لعميل

** المعلومات الخاصة بالمخزون :

تعد المعلومات التى تدور حول المخزون أكثر تعقيدا نوعا مـــا مـــن المعلومـــات الخاصة بالعملاء . فرغم أنه من السهل أن يتم تعريف السمات المرتبطـــة بصـــورة مباشرة بعنصر فى المخزن ، وذلك على النحو الموضح فى جدول (٣) ، إلا أن هناك حالات قليلة حيث توجد بعض المعلومات غير المرتبطة بصورة مباشرة . علـــى سبيل المثال ، ترتبط الأسئلة التى ترغب فى طرحها على العميل بشأن تحديـــد حالـــة اللعبة بنوع اللعبة ، وليس باللعبة فى حد ذاتها . يعنى ذلك أن كيان آخر يطلق عليـــه ToyTypes سوف يكون بحاجة الإحتفاظ بالمعلومات الخاصة بنوع اللعبـــة ، وذلـــك على النحو الموضح فى جدول (٤) وفضلا عن ذلك ، نظرا لوجود أســـئلة متعـــددة بشأن كل نوع من أنواع اللعب ، فسوف تكون بحاجـــة لكيـــان آخـــر (Condition Questions) كى ما يحتفظ بهذه الأسئلة ، وذلك على النحو الموضح فى جدول (٥)

جدول (٣)

العناصر الخاصة بالمخزون

اسم العمود	نوع البيانات	نوع VB	الوصف
InventoryID	Int	Long	أداة تعريف مميزة للعنصر الموجود في المجموعة
ToyTypeID	Int	Long	أداة تعريف مميزة لنوع اللعبة الموجودة في المجموعة
Name	Varchar(64)	String	اسم اللعبة
ManufacturerID	Int	Long	اسم المنتج الذي قام بصناعة اللعبة
YearIssued	Datetime	Date	التاريخ الذي تم فيه إنتاج اللعبة لأول مرة
Description	Varchar(256)	String	وصف اللعبة
MintValue	Money	Currency	قيممة اللعبة إذا كانت تخلو من أي عيوب
Condition	Int	Long	حالة اللعبة باستخدام معدل رقمي
ConditionMask	Varchar(64)	String	الإجابات الخاصة بالأسئلة المطروحة بصدد هذا النـــوع من اللعب

أسم العمود	نوع البيانات	نوع VB	الوصف
TrueValue	Money	Currency	القيمة الحقيقية للعبة طبقا للوضع الحالي
DatePurchased	Datetime	Date	تاريخ إضافة اللعبة إلي قائمة السلع الموجودة في المخزون
PurchasePrice	Money	Currency	سعر الشراء المطلوب في اللعبة
AskingPrice	Money	Currency	السعر المطلوب لبيع اللعبة، وتعني قيمة Zero أنك لا ترغب في بيع اللعبة أنذاك
BuyingPrice	Money	Currency	السعر المطلوب دفعه لشراء لعبة مماثلة
Wanted	Bit	Boolean	إذا كان صحيحا ، فإن ذلك يعني أنك ترغب في شراء اللعبة .
ForSale	Bit	Boolean	إذا كان صحيحا ، فإن ذلك يعني أنك ترغب في بيع اللعبة
Comments	Varchar(256)	String	أي تعليقات سوف يتم عرضها مع اللعبة
DateUpdated	Datetime	Date	تاريخ آخر تعديل تم إجراءه لهذه المعلومات

جدول (٤)

أنواع اللعب

اسم العمود	نوع البيانات	نوع VB	الوصف
ToyTypeID	Int	Long	أداة تعريف مميزة لنوع اللعبة الموجودة في المجموعة
Description	Varchar(64)	String	وصف نوع اللعبة

جدول (٥)

الأسئلة المتعلقة بتقييم حالة اللعبة

اسم العمود	نوع البيانات	نوع VB	الوصف
TypeID	Int	Long	أداة تعريف مميزة لنوع اللعبة الموجودة في المجموعة
Seq	Int	Long	الرقم التسلسلي الذي يتم استخدامه للتميز بين الأسئلة المتعددة الخاصة بنوع معين من اللعب
Question	Varchar(64)	String	السؤال المستخدم لتقييم حالة اللعبة
Weight	Int	Long	الأهمية النسبية للسؤال عند تقييم حالة اللعبة
Responses	Int	Long	أعلي قيمة ممكنة للإجابة

غير أنه في بعض الأحيان يكون من المفيد أن يتم تكويد قيمة البيانات لضمان وجود التناسق بين البيانات . وهناك مثال جيد لحقل يمكنه بسهولة الحصول علي بيانات غير صحيحة ألا وهو حقل Manufacturer . تصور كم الطرق المختلفة التي يمكن نطق Mattel من خلالها . لذا ، فهناك طريقة واحدة لضمان عدم حدوث خطأ في كتابتها وتتمثل في تكويد كل منتج لعبة بمثابة قيمة عددية وتخزين هذه القيمة العددية في قاعدة البيانات . عقب ذلك ، أنت بحاجة لإضافة جدول ترجمة يمكن الاستعانة به كي يتم ترجمة القيمة المشفرة إلي سلسلة أحرف من نوع نص ، الأمر الذي يتم تحقيقه من خلال جدول المنتجين ، وذلك علي النحو الموضح في جدول (٦)

جدول (٦)

المنتجون

اسم العمود	نوع البيانات	نوع VB	الوصف
ManufacturerID	Int	Long	أداة تعريف مميزة لاسم المنتج
Name	Varchar(64)	String	اسم المنتج

هناك مجال آخر جدير بالاهتمام يتمثل في الحاجة إلي تخزين صورة خاصة باللعب . فعلي الرغم أنني أعتقد أن تخزين الصور في قاعدة البيانات يعتبر أمراً مرضياً بالنسبة

إليك ، إلا أنني أرى كذلك أنه ينبغي تخزينها في جدول مستقل . ونظراً لأنني في حاجة إلـــي استخدام جدول مستقل للصورة ، فإنني اتخذت قراراً بإضافة عمود الرقم التسلسلي الذي مـن شأنه أن يتيح لي إمكانية تخزين صور متعددة للعبة واحدة ، وذلك علي النحو الموضح فـــي جدول (٧) .

جدول (٧)

الصــــــور

إسم العمود	نوع البيانات	نوع VB	الوصف
Inventoryld	Int	Long	أداة التعريــف ممــيزة للعنصــر الموجــودة فــى المجموعة .
Seq	Int	Long	الرقم التسلسلى الذى سوف يتم إستخدامة للتميــيز بين الصور المتعددة للعبة الواحدة
Imgee	Image	Picture	حقل ثنائى ضخم يحتفظ بالصورة الفعلية للعبة

** المعلومات الخاصة بالطلبيات :

يعتبر كيان Orders الكيان الرئيسى الأخير الذى قمت بتعريفه فى هذا التطبيق رغـم ذلك ، هناك حاجة إلى تقسيم هذا الكيان إلى أجزاء أصغر حتى يمكن أن يوجد لدى كل طلبيــة عناصر متعددة . من ثم ، يقوم جدول (٨) بسرد سمات كيــان Orders ، بينمـا يوضــح جدول (٩) السمات المرتبطة بعنصر واحد موجود فى الطلبية ولقد أطلقت على هذا الكيــان إسم OrderDetails .

جدول (٨)

الطلبيات

إسم العمود	نوع البيانات	نوع VB	الوصف
OrderId	Int	Long	أداة تعريف مميزة للطلبية
CustomerId	Int	Long	أدة تعريف مميزة للعميل
OrderType	Int	Long	1 = بيع ، 2 = شراء
Shipping Cost	Mony	Currency	تكاليف الشحن الإجمالية
SalesTax	Mony	Currency	التكاليف الإجمالية لضريبة المبيعات .
HowPaid	Int	Long	1= بطاقة الإئتمان ،2 = شيكات
CreditCardNumber	Varchar(32)	String	رقم بطاقة الإئتمان الخاصة بالعميل
ExpData	Varchar(16)	String	تاريخ إنتهاء بطاقة الإئتمان الخاصة بالعميل .
OrderStatus	Int	Long	1= إصدار طلبية ، 2= شحن الطلبية ، 3= إستلام الطلبية .
Date Orderd	Datetime	Data	تاريخ وزمن إصدار الطلبية
DataShipped	Datetime	Date	تاريخ وزمن وشحن الطلبية
DateReceived	Datetime	Date	تاريخ وزمن إستقبال الطلبية

جدول (٩)

كيان Order Details

الوصف	نوع VB	نوع البيانات	إسم العمود
أداة تعريف مميزة للطلبية	Long	Int	OrderId
رقم تسلسلى يتم إستخدامه للتمييز بين العناصر المتعددة فى الطلبية الواحدة .	Long	Int	Seq
أداة تعريف مميزة للعنصر الموجود فى المجموعة	Long	Int	InventoryId
سعر شراء اللعبة	Cu rrency	Mony	PuchasePric

أما الكيان الأخير الذى أرغب فى الحديث عنه الآن يتمثل فى كيان States . ويعد الغرض الأساسى من وجود هذا الكيان هو ترجمة إختصار الولاية المكون من حرفين إلى المعدل الخاص بضريبة المبيعات ، حيث يتم الإستعانة به لحساب مقدار ضرائب المبيعات المستحقة على الطلبية . وفى نفس الوقت ، لقد إتخذت قراراً بشأن إضافة حقل State Name لترجمة State إلى قيمة أكثر مدلولا.

جدول (١٠)

كيان States

الوصف	نوع VB	نوع البيانات	أسم العمود
إختصار ولاية مكون من حرفين	String	Char(2)	State
إسم الولاية	String	Varchar(64)	StateName
معدل ضرائب المبيعات الخاص بالولاية	Currency	Decimal	SalasTaxRate

ج - كيفية تعريف العلاقات :

لن يمثل تعريف العلاقات فى التصمم ثمة مشكلة إذا تبدأ العملية من خلال معرفة وتحديد كيفية ربط الكيانات مع بعضها البعض .

وهناك ثلاثة أنواع رئيسية من العلاقات : علاقة طرف بطرف وعلاقة طرف بأطراف وعلاقة أطراف بأطراف . وتشير هذه العلاقات إلى عدد إرتباط نسخ من البيانات فى كيان بنسخ من البيانات فى كيان آخر . فى علاقة طرف بطرف ، هناك نسخة واحدة فحسب من البيانات فى كيان مرتبطة بنسخة واحدة من البيان فى كيان آخر . على سبيل المثال ، إفترض أنه يوجد لديك كيانان – المتاجر والمديرين . يدار كل متجر من خلال مدير واحد فحسب ، وفى الوقت نفسه يقوم المدير بإدارة متجر واحد فحسب . من ثم يعنى ذلك الأمر أن كل متجر يوجد لدية مدير معين وكل مدير يوجد لديه متجر محدد .

فى علاقة طرف بأطراف ، ترتبط نسخة واحدة من البيانات فى الكيان الأول بصفر أو أكثر من نسخ البيانات فى الكيان الثانى . على سبيل المثال ، إفترض أنه يوجد لديك كيان للعملاء وكيان آخر للطلبيات . قد يقوم كل عميل بإصدار العديد من الطلبيات حسب الرغبة . غير أن العميل ليس فى حاجة لإصدار أية طلبيات فى حالة ما إذا قام بتسجيل إسمه فى قائمة بريدية . إما فيما يتعلق بكل طلبية ، فهناك عميل واحد هو الذى قام بإصدار الطلبية . من ثم هناك عميل واحد فحسب لكل طلبية ، على حين قد يوجد لدى كل عميل طلبيات متعددة أو لايوجد لدية ثمة طلبيات على الإطلاق .

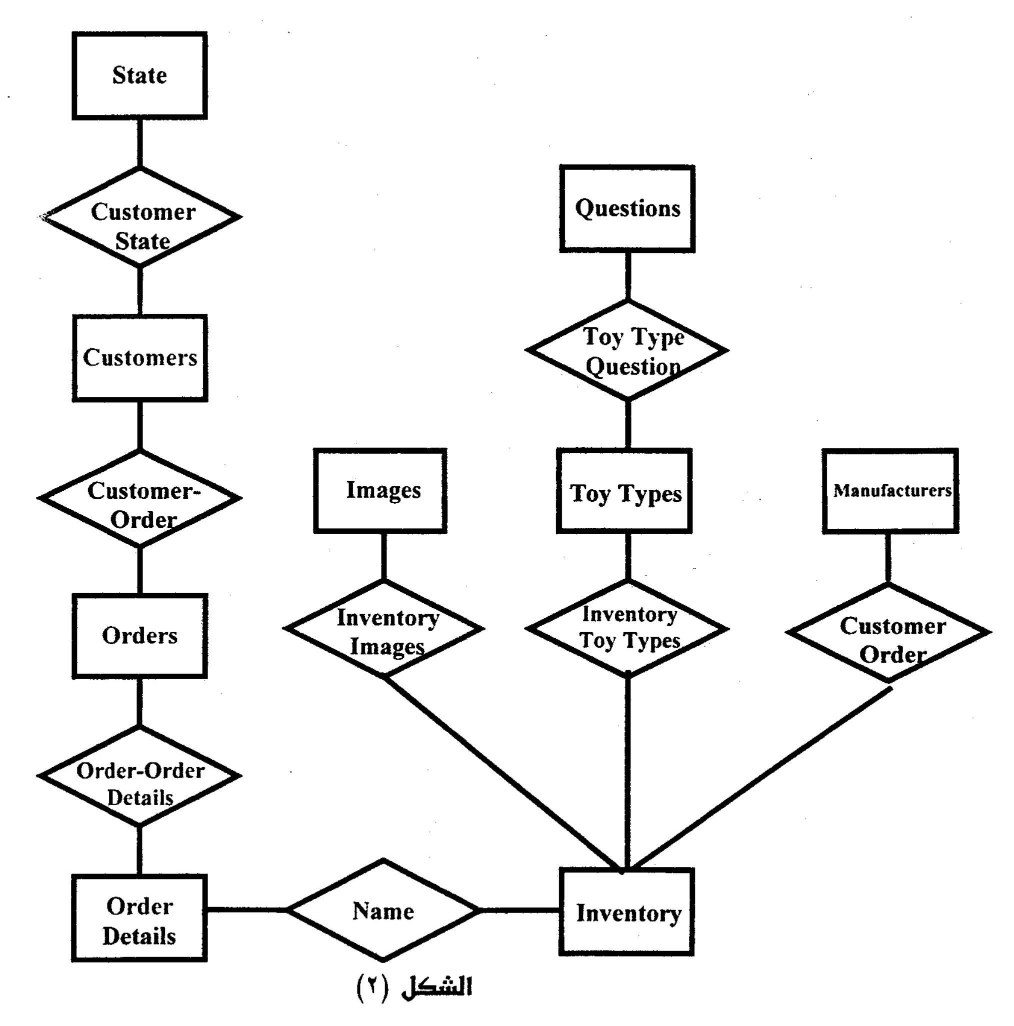
فى علاقة أطراف بأطراف ، ترتبط نسخ متعددة من البيانات فى الكيان الأول بنسخ متعددة من البيانات فى الكيان الآخر . ويمكن تصوير ذلك من خلال وجود كيان للأسر وكيان آخر للأبناء . قد يوجد لدى كل أسرة عدد من الأبناء وقد لايوجد لديها أى أبناء وفى الوقت نفسه قد يوجد لدى الطفل أكثر من أسرة (تذكر ، الطفل اليتيم لايوجد لدية أية أسرة ، أما الطفل الذى إنفصل أباه عن أمه قد يوجد لدية أو وأب وزوج أم زوجه أ ب)

د- كيفية رسم نمط E/R :

تعتبر عملية تصميم نمط E/R مهمة سهلة للغاية ويتم ذلك من خلال المعلومات المتوفرة فى الجداول بدء من جدول (٢) ووصولا إلى (١٠) ، وذلك كما هو موضح فى شكل (٢) . رغم أنى لم أوضح السمات الخاصة بكل كيان نظرا لأن ذلك من شأنه أن يجعل

الرسم الصغير غير مقروء تقريبا ،إلا أن الأمر فى غاية السهولة . بالإضافة إلى ذلك ، فإن مقارنة الجداول المذكورة أعلاه مع الشكل التوضيحى سيكون بالطبع أكثر مدلولا أيضا .

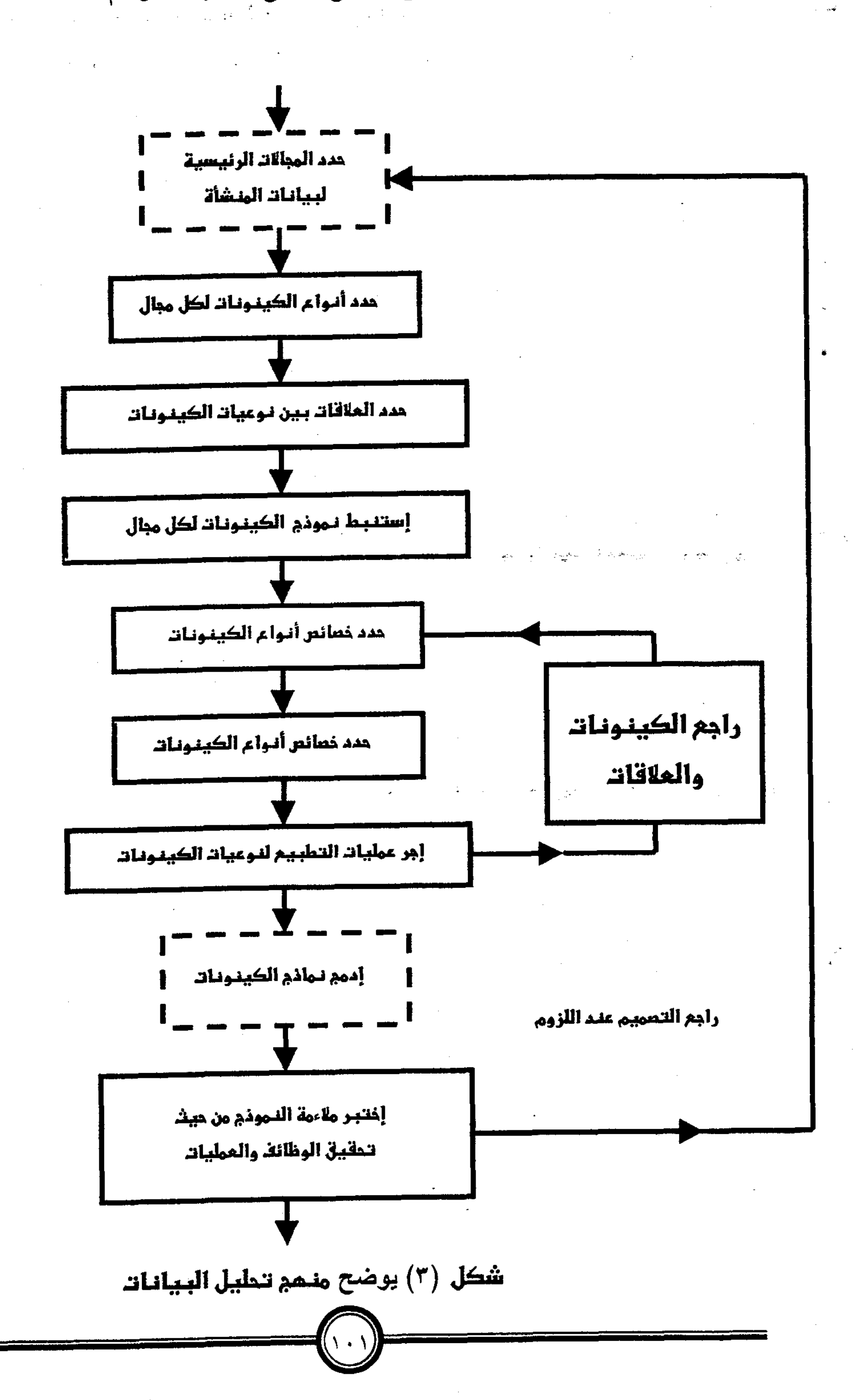
ملحوظة : عند رسم نمط E/R ، فإننى أقترح إستخدام أداة مثل Visio بدلا من إنشاء الرسم بالورقة والقلم . إذ يتيح لك برنامج Visio إمكانية تحرير الرسم بسهولة كى ما يتوافق مع التغييرات الحتمية التى تحدث أثناء إستعراض المستند الخاص بك والتعليق عليه من قبل الكثيرين . ومما لاشك فيه أن هناك بعض أداوت تصميم قواعد البيانات الباهظة للغاية والتى تعرض إمكانيات مماثلة إلا إننى أعتقد أن Visio يعمل جيدا تقريبا بالنسبة لمعظم تصميمات قواعد البيانات .



الشكل (٢)

عرض نمط Entity/Relationship النهائي الخاص بقاعدة بيانات Toy Collector

ويمكن لنا أن نوجز فيما يلى المنهج المتبع عند إعداد رسم ER



شكل (٣) يوضح منهج تحليل البيانات

(١) **حدد المناطق الرئيسية للبيانات** : عادة ما يتم التعرف على هذه المناطق من خلال وصف المنظمة الرئيسية ، ومن أمثلتها معالجة طلبيات العملاء ، التسويق ، المشتريات ... إلخ .

(٢) **تعرف على أنواع الكينونات فى كل منطقة** : ليس لهذه المهمة قواعد حاسمة ، ولكن يمكن التعرف عن كون شئ ما يجب إعتباره نوعا لكينونة من خلال السؤال التالى : هل تود المنشأة الإحتفاظ ببيانات عن هذا الشئ لأى سبب من الأسباب ؟

(٣) **حدد العلاقات بين الكينونات** : لايجب على المحلل فى هذه المرحلة أن يكتفى بالعلاقات القائمة بالفعل ، بل عليه أن يضع فى الإعتبار الإحتمالات واردة الحدوث . نفرض أن العلاقة بين الموردين والأصناف المخزنية هى حاليا 1 : ن بمعنى أن كل مورد لا يورد إلا صنفا واحدا مثل هذه العلاقة قد تتطور فى المستقبل إلى م : ن ، حيث يحتمل أن يقوم مورد بتوريد أكثر من صنف . فمن الصواب أن يعتبر المحلل العلاقة من هذا النوع .

(٤) **قم ببناء النموذج** : يضم النموذج الكينونات والعلاقات بينها . ويقوم المحلل برسم هذه النماذج على الصورة المبينة فى شكل (٤) .

(٥) **حدد خواص الكينونات** : تقابل هذه الخواص البيانات التى سيحتفظ بها عن كل كينونة ويراعى أن تكون هذه الخواص شاملة بقدر الإمكان ، ولكن الواقع العملى يبين أنه يجرى دائما التعديل فى هذه الخواص خلال التصميم .

(٦) **قم بعمليات التطبيع اللازمة** : تهدف عملية التطبيع Normalization إلى التأكد من النموذج يحتوى على الكينونات فى أبسط صورها . ويتم ذلك بتحليل الكينونات كينونات أبسط إلى أن نصل للكينونات الأولية التى لاتقبل التحليل إلى مكونات أبسط .

(٧) **إدمج النماذج فى نموذج شامل للمنشأة** .

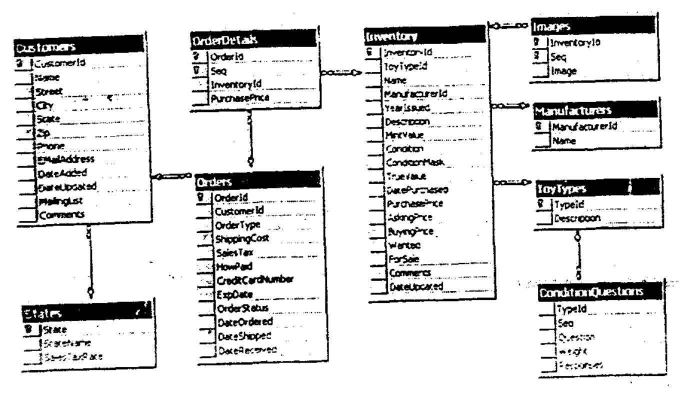
(٨) **إختبر مدى ملاءمة النموذج** : كيف يتأكد المحلل من أن النموذج الموضوع سوف يفى بالغرض ؟ وهناك ثمة سؤالان يجب الإجابة عنهما :

هل ستكون البيانات المطلوبة موجودة ؟

هل يمكن الوصول للبيانات ؟

٤- كيفية بناء قاعدة البيانات :

تعتبر عملية ترجمة نمط E/R إلى قاعدة بيانات عملية سهلة للغاية . إذا يضحى كل من الكيانات جدولا ، كما تصبح السمات الخاصة بالكيان أعمدة فى هذا الجدول ويمكن مشاهدة المنتج النهائى فى شكل (٤) بإستخدام الشكل التوضيحى لقاعدة بيانات SQLServer.



الشكل (٤)

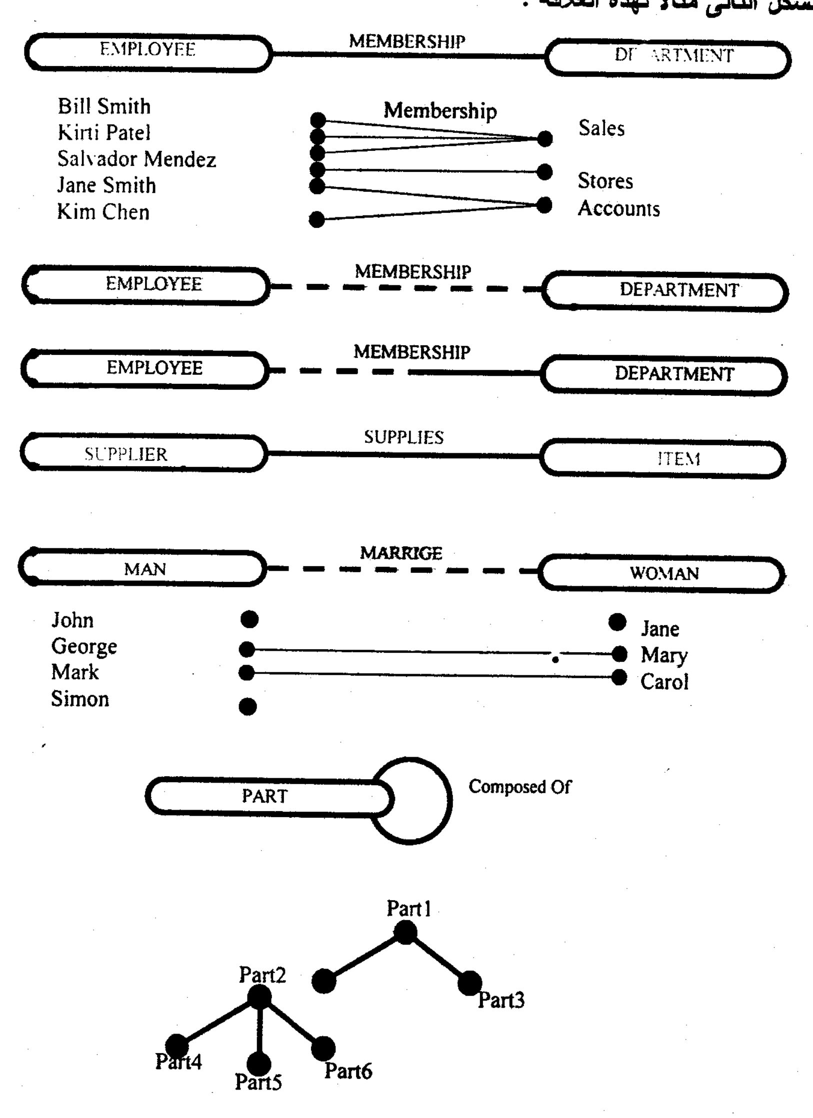
الشكل التوضيحى الخاص بقاعدة بيانات Toy Collector

٥- إنشاء التطبيق :

وهذه تعد آخر خطوة ، حيث أنه فى ضوء ماسبق يمكن الوصول الى التصميم المنطقى لشكل التطبيق الذى سيستخدم فى هذا الصدد .

العلاقات الملتفة Involuted relation

يقصد بالعلاقات الملتفة علاقة وقائع الكينونة ذاتـــها ببعضـــهم البعـــض ، فبعـــض الأصناف المخزنية تجمع من أصناف مخزنية أخرى ويمثل ذلك شكل (٥١) ، وفيـــه يوجـــد جزء منقط يشير لضرورة وجود بعض الأصناف الأولية التى لاتجمع مـــن أصنـــاف أخـــرى ويبين الشكل التالى مثالا لهذه العلاقة .



شكل (٥) أنواع العلاقات

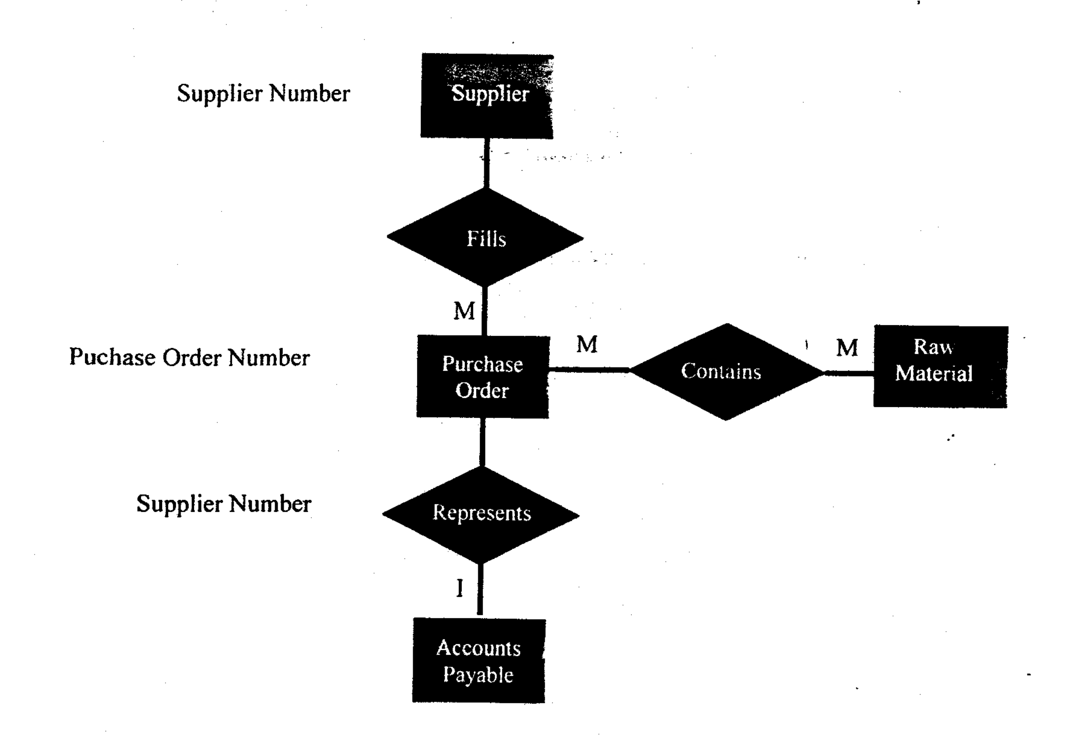
مثال توضيحى لرسم كينونة - علاقة Entity realatioship Diagram(ERD)

فيما يلى مثال يوضح رسم كينونة – علاقة تقريبيه للبيانات المشمولة فــى شــراء مواد من الموردين . التى تطلب مواد خام تحتاجها المنشأة وسوف تأخذ هذه العملية ســبعة خطوات أهمها

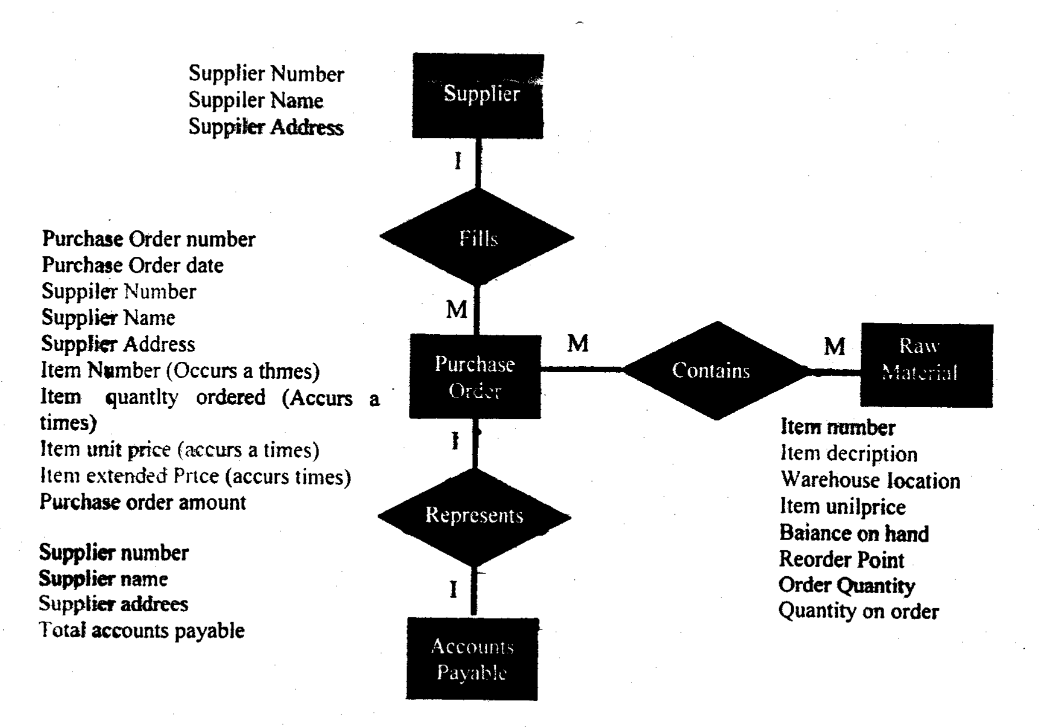
(١) **تعريف الكينونات** . تقرر الإدارة أى العناصر البيئية ، والموارد ، والعمليــات الجارية سوف توصف مع البيانات .

(٢) **تعــريف العلاقات** : ترتبط كل كينونة بكينونة أخرى بنوع معين من الإجراء

(٣) **إعداد رسم كينونة - علاقة تقريبى** . تخطط الرموز بحيث تقرأ العلاقــات مــن اليسار إلى اليمين أو من أعلى إلى أسفل كلما كان ذلك ممكنا . ويبيــن شــكل (أ-٢) مثالا لرسم كينونة – علاقة تقريبى للبيانات المشمولة فى شراء مــواد من الموردين . فيلبى المورد أوامر الشراء التى تطلب مــواد خــام تحتاجــها المنشأة فى شكل (٦) رسم كينونة – علاقة تقريبى .



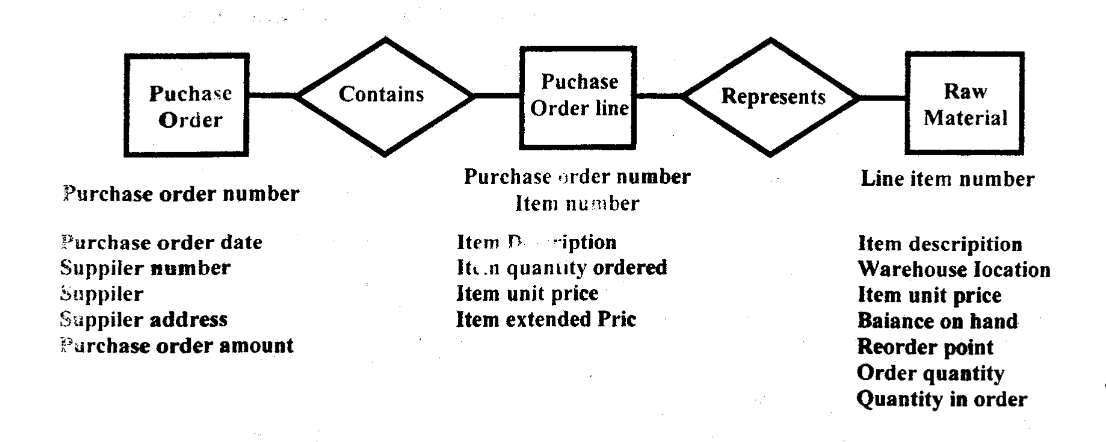
(٤) تحديد عناصر بيانات للكينونات : تسرد عناصر البيانات التى تعرف وتصف كل كينونة بيانات بعد الكينونة ، كما هو مبين فى شكل (٧) .



شكل (٧) تحدد عناصر البيانات لأنواع الكينونات الخاصة بها

(٥) **تنفيذ تحليل البيانات** : تدرس عناصر البيانات بغرض جعل هيكل قاعدة البيانات كفؤا . وتسمى عملية إجراء تحليل البيانات تطبيع Normaliztion ، والمهمة عبارة عن تضبط البيانات بحيث إنها تترجم بسلسلة من الصيغ الطبيعية .

أول صيغة طبيعية : الغاء العناصر المكررة فى الكينونة . عندما تكون البيانات فى أول صيغة طبيعية ، لايحدث أى عنصر بيانات مرآت متعددة لكينونة معينة . مثال ذلك يجب ألا يحتوى سجل أمر شراء على عناصر مطلوبة متعددة . لاحظ الترميز " حدث n مرة " فى شكل (٧) تلغى هذه المشكلة عن طريق إنتاج كينونة جديدة ، مثل سطر أمر الشراء ، كما هو مبين أدناه . فى هذا المثال ، تحتوى الكينونة الجديدة على خاصيتين معرف – رقم أمر الشراء ، ورقم العنصر . وتسمى مثل الخواص متعددة المعرف هذه مفتاحا مركبا Composite Key . وتحتوى الكينونة الجديدة على العديد من الخواص الواصفة أيضا ، إلا أنها تحدث مرة واحدة فقط – مرة واحدة فى سطر العنصر وقد نقلت خاصية الوصف فى كينونة سطر أمر الشراء من كينونة أمر الشراء .



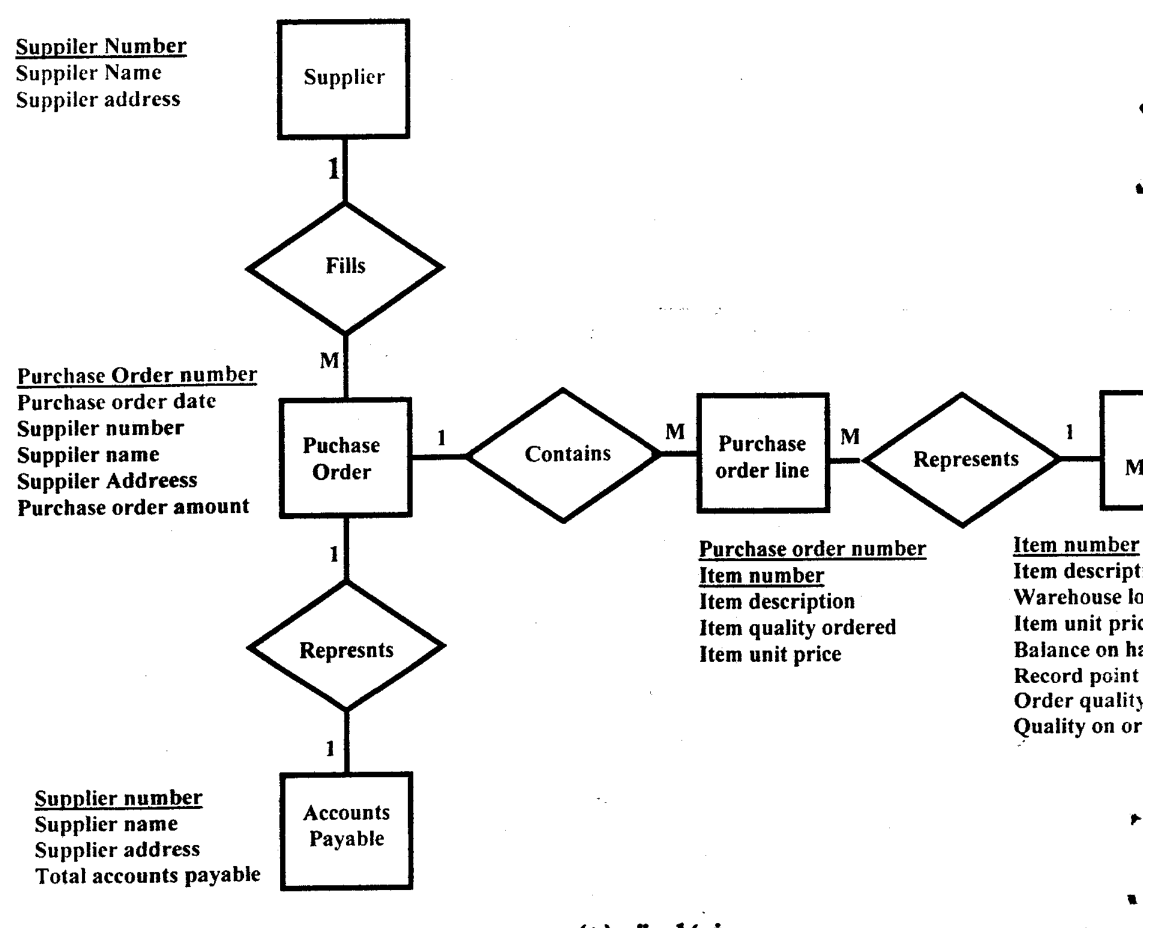
ثانى صيغة طبيعية : ضمان أن خواص الواصف تعتمد على مفتاح التكوين الكلى لتحديد القيم . تتحدد القيم . تتحدد قيم الخواص لكل حدوث كل كينونة . وعندما يكون للكينونة مفتاح مركب ، يمكن أن يحتاج تحديد قيمة لخاصية واصف محدد إلى كل من خاصيتى المحددين للمفتاح المركب . مثال ذلك ، تحتاج كمية العنصر التى تطلب فى سطر أمر الشراء إلى تحديد كل من أمر الشراء ورقم العنصر . وتحتاج خواص الواصف الأخرى إلى أحد خواص المعرفات فقط . فيحتاج وصف العنصر لأحد العناصر المطلوبة فى أمر الشراء إلى رقم العنصر فقط . وفى هذه الحالة ، يكون رقم أمر الشراء غير مناسب وفى المثال السابق ، تعتمد كمية العنصر المطلوبة وسعر الإتساع للعنصر على المفتاح المركب كله فقط . لذلك ، يمكن حذف خواص الواصفات الأخرى (وصف العنصر وسعر الوحدة للعنصر) . وعند طباعة أحد الأسطر فى أمر الشراء ، يمكن الحصول على وصف العنصر وسعر الوحدة للعنصر من كينونة المادة الخام .

ثالث صيغة طبيعية : ضمان أن قيمة الخاصية لايمكن تحديدها بقيم خواص أخرى . يحدث هذا الموقف عندما تحسب قيمة الخاصية من خواص أخرى . ففى المثال السابق يمكن حساب سعر الإتساع للعنصر عن طريق حسابة بضرب سعر الوحدة للعنصر (من كينونة المادة الخام) فى الكمية المطلوبة (من كينونة سطر أمر الشراء) . ونظرا لإمكانية حساب سعر الإتساع للعنصر ، فيمكن حذفه كخاصية واصف لكينونة سطر أمر الشراء .

(٦) **إعداد رسم كينونة -علاقة معدل** : تدخل نتائج تحليل البيانات فى رسم كينونة – علاقة جديد كالمبين فى الشكل (٨) . وبهذه الطريقة ، ترتب أنواع الكينونات وعلاقاتها بحيث توفر الأساس الأكثر كفاءة لتصميم قاعدة البيانات .

(٧) راجع رسم الكينونة - العلاقة مع المستخدمين وقم بتقنية يراجع المتخصص فى المعلومات الرسم مع منفذى الإدارة العليا ، والمديرين ، وغير المديرين فى منطقة المستخدمين وينقية كلما كانت هناك حاجة لذلك .

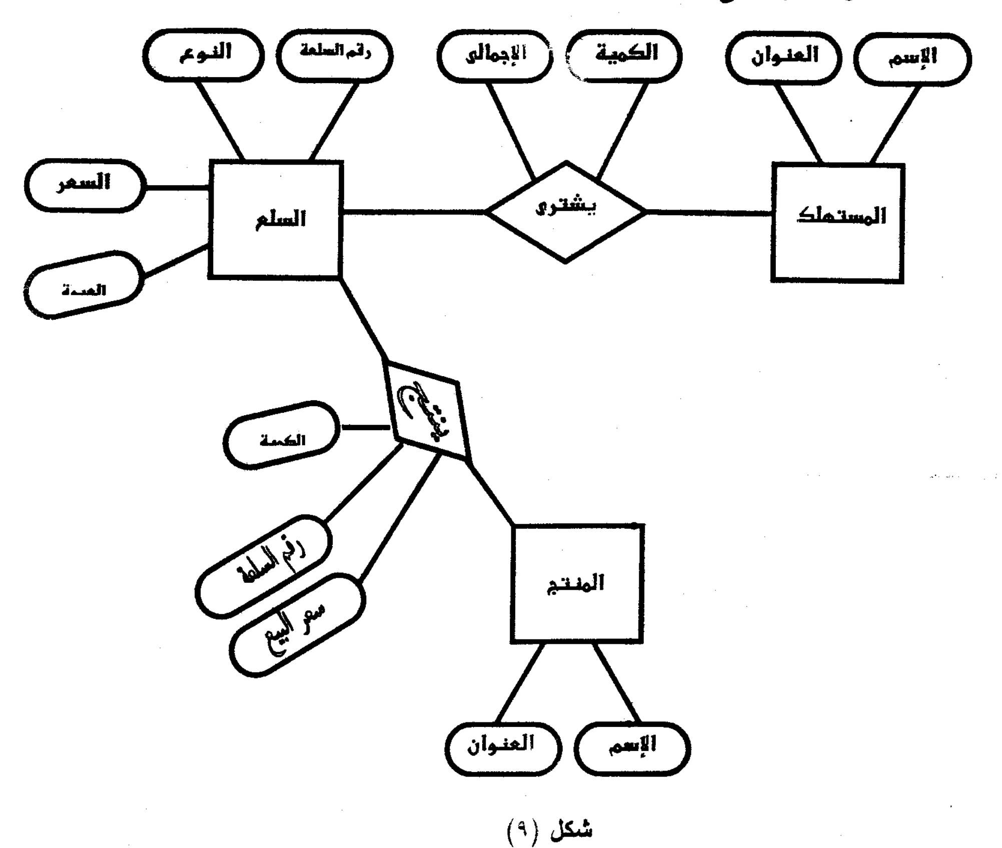
ويوفر رسم الكينونة - العلاقة الصورة الكبيرة لبيانات المنشأة . ويجعل قاموس البيانات هذه الصورة أكثر إكتمالا .



شكل رقم (٨)

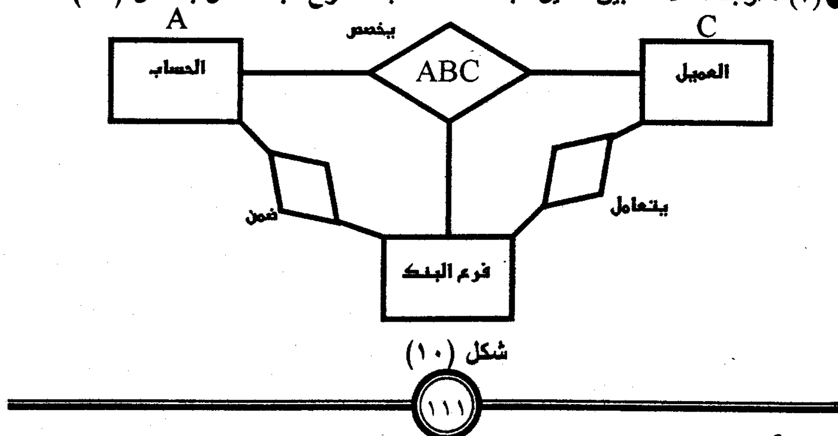
أمثلة توضيحية متنوعة
على خرائط العلاقات - الكينونات

مثال(١) : ارسم نمط ERD تتضمن الكيانات التالية : المستهلك – المنتـــج – السلع مع تحديد مجموعات الخصائص المرتبطة بكل كينونة .يمكن تحديد ذلك من خلال الشكل التالى :



شكل (٩)

مثال(٢) : أوجد العلاقات بين عميل البنك – الحساب – فرع البنك الحل بالشكل (١٠)



شكل (١٠)

ونلاحظ من الأمثلة السـابقة مـايلى :

أ-لكل كيان دور فى العلاقة .

ب- قد تكون العلاقة فى إتجاه واحد أو إتجاهين ، لكل مواطن .. تليفون وبالتـالى يمكن لإسم صاحب التليفون تحديد رقمه ، كما يمكن من رقم التليفـون تحديـد إسم صاحبة وعنوانه .

ج- معظم ما عرضنا حتى الآن علاقات إحادية ١ : ١ مثلا للعميل الواحـد حسـاب وحيد فى فرع البنك ، وكل حساب مخصص لعميل واحد .

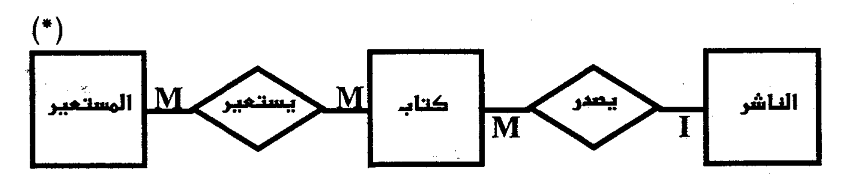
الإرتبـاط المتعـدد :

توجد ثلاث إحتمالات للارتباط بين العلاقات :

أ- علاقة واحد إلى واحد [١: ١] كما أشرنا فى الأمثلة السابقة .

ب- علاقة واحد إلى متعدد [١:م] وتكتب [1 : M] حيث M تعتـبر عن Many أو Muliple .

ج- علاقة متعدد إلى متعدد [م : م] أو [M :M] ويوضح الشكل (١١) مفـهوم التواجد المتكرر للكيانات وكيفية إدراجها فى ملفات النظام .



(*) الملف بالشكل (١٢)

شكل (١١)

رقم المستعير	الإســـم	الدرجة	القسم	رقم التليفون الداخلى
٦٦٨	محمد	١	الانتاج	١١٦
٧٩٤	هناء	٢	التسويق	٩١٣
٢٥٦	[illegible]	٣	المالى	٢١١
٦٧١	منال	٤	الأفراد	٢٥٦
٦٢٣	جرجس	٥	ش-قانونية	٣٠٧
٢٣٢	إيمان	٦	علاقات	٤٠٨

التواجد المتكرر للمواصفات

شكل (١٢)

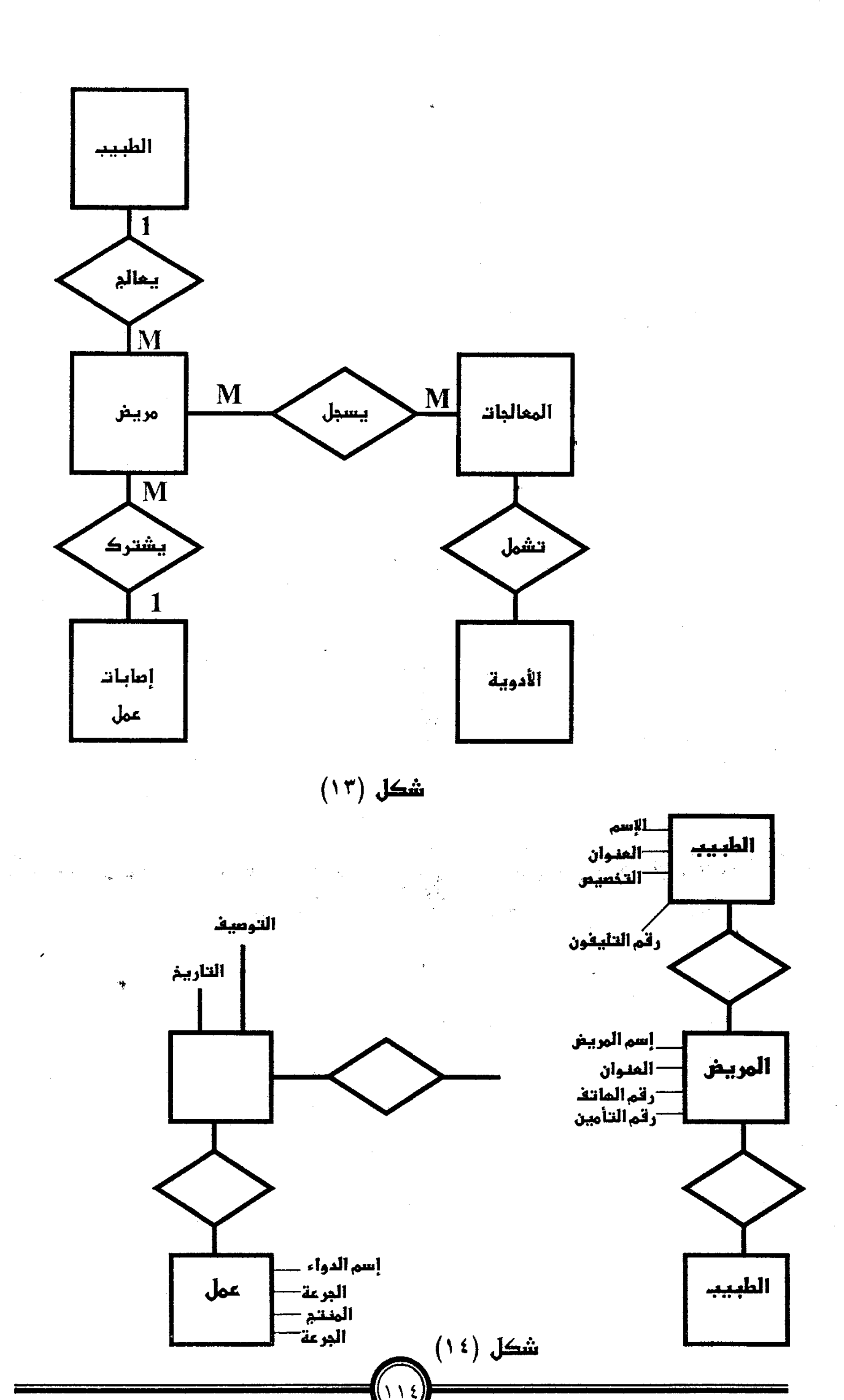
مثال (٣) : إرسم وحدد عناصر البيانات اللازمة للعلاقة بين الطبيب والمريـــض وروشـــتة العلاج ؟ علما بأن المرضى مؤمنون ضد إصابات العمل؟ [ملحوظة : راجـــع مثال مستشفى إيهاب فى الباب السادس]

الحل :

تعتـــبر ERD المثـــال مـــن النـــوع متعـــدد الكيانـــات ودرجـــة العلاقـــة أيضـــا CARDINALLTY بها تنوع كبير للأسباب التالية :

أ- الطبيب يعالج عدد (متعدد) من المرضى :

ويفسر هذا على أنه من المستحيل تواجد طبيب لكل مواطن حتـــى فـــى أرقـــى دول العالم حيث لاتقل النسبة عن طبيب لكل ٥٠٠ مواطن .



الطبيب
1
يعالج
M
مريض
M
يسجل
M
المعالجات
M
يشترك
1
إصابات عمل
تشمل
الأدوية

شكل (١٣)

الطبيب
الإسم
العنوان
التخصيص
رقم التليفون
المريض
إسم المريض
العنوان
رقم الهاتف
رقم التأمين
الطبيب
التوصيف
التاريخ
عمل
إسم الدواء
الجرعة
المنتج
الجرعة

شكل (١٤)

ب- كل مريض له رقم تأمين ضد إصابات العمل .

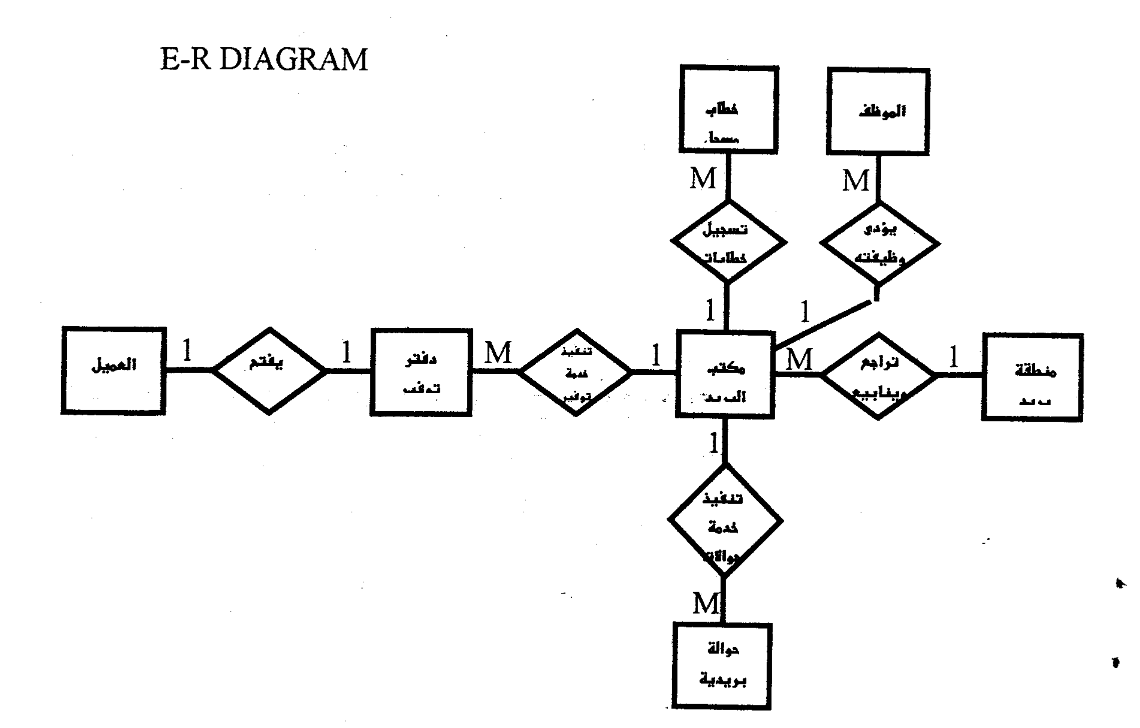
ج- كل مريض يتطلب رعاية لاحقة من الطبيب تستدعى تسجيل بيانات عنه .

د- يحتمل مع كل مراجعة للطبيب وتوصيف وكتابة أدوية جديده أو معالجـــة طبيـــة مختلفة

هـــ- يوضح الشكل (١٣) خريطة ERD.

و- يوضح الشكل (١٤) موصفات كيانات الشكل (١٣) .

مثال(٤) : العلاقات بين الكيانات فى نظام تطوير مكاتب البريد :



مثال (٥) :

تتولى شركة الأصيل التجارية عملائـــــها باستخدام مندوبي مبيعات يعرضون على العملاء إنتاج الشركة ، ويقوم العملاء بتحرير أوامر الشراء وإرسالها أو تسليمها إلى المندوبين والذين بدورهم يرسلونها إلى الشركة حيث تتولى إعداد الأصناف المطلوبة وإرسالها للعملاء . والمطلوب إعداد قاعدة بيانات عن هذا النشاط .

الحل :

١- يتم تحديد كيانات النظام وهي :

أ- مندوب البيع .

ب- العميل .

ج- أمر الشراء .

د- الأصناف

٢- يتم تحديد خصائص الكيانات :

أ- مندوب البيع (إسم المندوب – رقم المندوب – منطقة عمل المندوب) الحقل المفتاح رقم المندوب .

ب-العميل (رقم العميل – الإسم – سجل العميل – رقم التليفون – الحالة) الحقل المفتاح رقم رقم العميل .

ج- أمر الشراء (رقم الأمر – التاريخ – رقم الصنف – وصف الصنف- العدد المطلوب- الحقل المفتاح : رقم الأمر).

د- الصنف (رقم الصنف – الكمية المتاحة – وصف الصنف) الحقل المفتاح – رقم الصنف .

هـ- النماذج (...) الحقل المفتاح رقم المخزن .

٣- **دراسة العلاقات ومنها يتضح مايلى :**

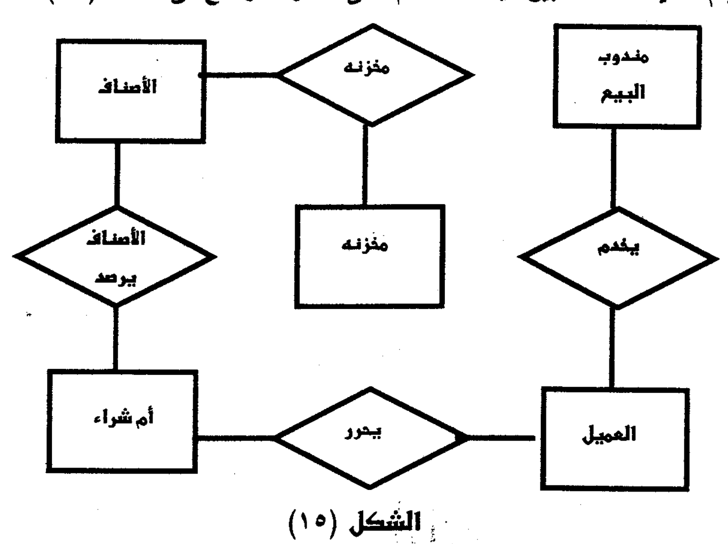
أ- العلاقة بين مندوب البيع والعميل القوية .

ب- باقى العلاقات ضعيفه .

٤- **إعداد نمط رسم ER للكيونات والعلاقات السابقة .**

٥- **إنشاء ملفات لخصائص الكيانات والعلاقات فى شكل جداول .**

يتم تحديد العلاقات بين كيانات النظام على النحو الموضح فى الشكل (١٥)



الشكل (١٥)

- يعبر المربع عن كيان باعث للبيانات وينشأ له ملف .
- يوصف الكيان باسم (مندوب البيع – العميل – أمر شراء ... إلخ) .
- تنشأ العلاقة بين الكيانات نتيجة حركة ومشاركة بيانات وتوصف بفعل (يخدم – يحرر – يرصد الأصناف ...) .
- قد تكون العلاقة بين كيان وكيان واحد أو عدة كيانات فمندوب البيع يخدم أكثر من عميل وتكون العلاقة فى هذه الحالة واحدة إلى متعدد ، كما يحتمل أن تكون العلاقة أطراف إلى أطراف .

رقم الأمر	التاريخ	رقم المنتج	الوصف	الكمية
٤٠٣١١	١٩٩٣/١١/٣	٣٠٤	أ	٧
		٦٣٣	ب	١
		٦٨٤	ج	٤
٤٠٣١٢	١٩٩٣/١١/[illegible]	١٢٨	٥	١٢
		٣٠٤	أ	٣

من دراسة (الجدول) السابق نلحظ وجود تكرارات فى رقم السلعة ووصف السلعة مما يستدعى إزالة هذ التكرارات بفرض رقم الأمر على كل سلعة فى المجموعات المتكررة ليصبح الجدول على الشكل التالى :

رقم الأمر	التاريخ	رقم السلعة	الوصف	الكمية
٤٠٣١١	١٩٩٣١١٠٣	٣٠٤	أ	٧
٤٠٣١١	١٩٩٣١١٠٣	٦٣٣	ب	١
٤٠٣١١	١٩٩٣١١٠٣	٦٨٤	ج	٤
٤٠٣١٢	١٩٩٣١١٠٣	١٢٨	د	١٢
٤٠٣١٢	١٩٩٣١١٠٣	٣٠٤	أ	٣

ورغم هذا الإجراء فلا زالت هناك تكرارات فى الجدول مما يدعو إلى القول أن لا رقم الأمر أو رقم السلعة يمكن أن يكونا رقم وحيد كل منفردا عن الآخر مما يستدعى ضم الرقمين وتكوين حقل مركب ذى رقم وحيد لا يتكرر :

رقم الأمر	التاريخ	الوصف	الكمية
٣٠٤٤٠٣١١	١٩٩٣١١٠٣	أ	٧
٦٣٣٤٠٣١١	١٩٩٣١١٠٣	ب	١
٦٨٤٤٠٣١١	١٩٩٣١١٠٣	ج	٤

ويسمى ما اجريناه هو التبسيط رقم (١) **first normal form**

بأخذ النظام الفرعى الخاص بمعالجة طلبيات العملاء بشركة " أبو الفتــوح " مثــالا على العلاقات الملتفة يتم إتباع ما يلى :

١) **تحديد أنواع الكينونات** : يقترح المحلل الأنواع التالية المكينونات المتعلقة بالنظــام الذى يتعامل معه.

** العميل – طلبية العميل – نموذج الشركة – الصنف – كتــالوج المبيعــات – البضاعة قسيمة الشحن – الفاتورة .

** وبإجراء مراجعة هذه الكينونات يتضح الآتى :

** طلبية العميل يسجل فى نموذج الشركة ، ومن ثم فتستبعد هذه الكينونة بشــرط التأكد من كافة بياناتها مسجلة فى النموذج .

** الصنف والبضاعة مصطلحين مترادفين ولذا تستبعد " البضاعة "

** البيانات الموجودة فى كتالوج البضاعة هى خصائص " الصنف " ومن ثــم فــلا تعتبر كينونة ذات علاقة بالمنطقة التى يجرى تحليلها بينما قد يعتبر كينونة فــى عملية أخرى (عملية تحديث كتالوجات الشركة).

بناءا على ماتقدم – يستقر على الكينونات التالية وعلى صفاتها المفتاحية التالية :

العميل (رقم العميل) – نموذج الشركة (سيطلق عليــة " طلبيــة للتبســيط) (رقــم الطلبية) – الصنف (رقم الصنف) – قسيمة الشحن (رقــم القســيمة) – الفــاتورة (رقــم الفاتورة) .

٢) **إيجاد العلاقات بين الكينونات** : يمكن من خلال تقييم دراسة الحالة تحديد العلاقــات التى تربط بين أنواع الكينونات ، ومن المفــترض أن تكــون العلاقــات متمثلــة فــى العلاقتين التاليتين :

يقدم العميل أكثر من طلبية ولاتقدم الطلبية إلا من عميل واحد . العلاقة 1 : ن

الطلبية تتضمن عدة أصناف ، والصنف يكون لعدة طلبيات ، العلاقة م : ن

ويبين جدول (١٠) العلاقات بين الكينونات :

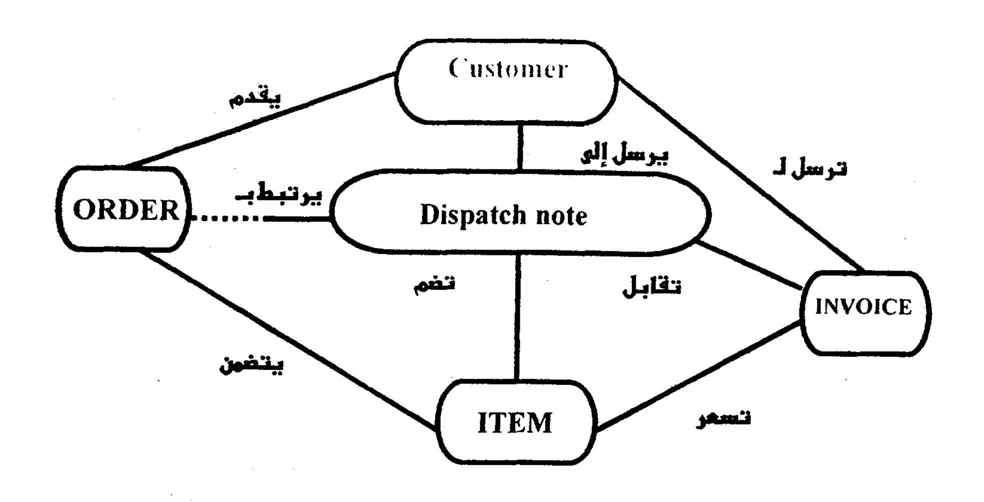
ENTITY	ENTITY	RELATIONSHIP	DEGREE
CUSTOMER	ORDER	PLACES	1:n
ORDER	ITME	REQUESTS	M:N
DESPATCHER NOTE	ITEM	COMPRISES	M:N
DESPATCHER NOTE	ORDER	RELATES	N:1
INVOICE	ITEM	Bills	M:N
DESPATCHER NOED	INVOICE	Relates to	1:1
DESPATCHER NOTE	CUSTOMER	SENT TO	N:1

[ونقدم فيما يلى جدولا مقابلا باللغة العربية للمزيد من الإيضاح ، أخذا فى الإعتبار أن الترجمة للعلاقات ليست حرفيه] .

الكينونة	الكينونة	العلاقة	نوع العلاقة
عميل	طلبية	يقدم	1 : ن
طلبية	صنف	يتضمن	م : ن
قسيمة شحن	صنف	تضم	م : ن
قسيمة شحن	طلبية	ترتبط بـ	ن : 1
فاتورة	صنف	تسعر	م : ن
قسيمة شحن	فاتورة	تقابل	1 : 1
قسيمة شحن	عميل	ترسل لـ	ن : 1

ويجب أن تكون مسميات العلاقات معبرة عن معناها ، وقد تثـــور مشــكلة حينمــا يصعب إيجاد إسم مناسب أو يتطلب الأمر إطلاق نفس الإسم على علاقتين ، ويخضع التعـامل فى هذه المشاكل لخبرة المحلل وحسن تصرفه .

٣) **بناء النموذج** : يوضح شكل (١٦) مخطط علاقات الكينونات ، وهـــو يصــور أنــواع الكينونة وطبيعة العلاقات بينها كما وردت فى الجدول (١٠) .



شكل (١٦) مخطط علاقات الكينونات لعملية تنفيذ طلبيات البيع

معنى الخط المنقط فى العلاقة " تتضمن " بين الصنف والطلبية أنه من الممكن ألا تتضمن بعض البيانات بعض الأصناف . ونفس الشئ يقال عن العلاقة " تسعير " بين الصنف والفاتورة العلاقة تضم بين الصنف وقسيمة الشحن ، حيث أنه بالنسبة للصنف الذى يتضمن فى الطلبية لن تتضمن السعر ولن تضمه قسيمة الشحن .

كثيرا ما يمكن دمج الكينونات ذات العلاقات 1 : 1 فى كينونة واحدة ، ولكن لم يطبق ذلك مع العلاقة بين قسيمة الشحن والفاتورة لكونهما مستقلين ويدخلان فى علاقات مختلفة مع الكينونات أخرى .

٤) **تحديد خصائص الكينونات** : يوضح الجدول (١١) خصائص كافة الكينونات التى يضمها التحليل .

جدول (١١) خصائص الكينونات

Entity	Attribute
OEDER	Order # Order Data Customer # Customer Name , [item — #Item Quantity , item Price]*,
CUSTOMER	Customer # Customer Name , Customer address , Ideliven Address]*, credit [limit balance,Turnover year To data .
DESPATCH NOTE	Despatch # despatcher data , order # Customer #Customer name , delivery address , [item#Quantity
Invoige	Invoice#,Invoice data , order # , Customer name customer address , [item #, item description,item price item quantity]*, subtotal sales tax , total payable.
ITEM	item #, item description quantity in stock , item price re-order level .

الشـــرح :

الكينونة	الخصائص
طلبية	رقم التاريخ ، رقم العميل ، [رقم الصنف ، الكمية ، السعر].
عميل	رقم ، إسم ، عنوان ، [عنوان التسليم] * ، رقم التليفون ، حد الإئتمان ، جملة المبيعات السنوية حتى تاريخه Turnover Year data ، الرصيد الحالى Current balance .
قسيمة شحن	رقم ، التاريخ ، رقم الطلبية ، رقم العميل ، إسم العميل ، عنوان التسليم، [رقم الصنف ، الكمية] *
فاتورة	رقم ، التاريخ ، رقم الطلبية ، رقم العميل ، إسم العميل ، عنوان العميل [رقم الصنف ، الكمية ، السعر] * ، ثمن البضاعة ، الخصم ، ضريبة المبيعات ، صافى القيمة .
صنف	رقم الصنف ، إسم الصنف ، الكمية المتاحة ، السعر ، حد الطلب .

ويلاحظ من الجدول الخصائص المتكررة والتى توضع بين مربعين وتميز بنجمة .
وفى تلك المرحلة قد تظهر الحاجة لتعريف كينونات جديدة ، ويجب أن تكون الخصائص معرفة فى قاموس البيانات .

٦) **تطبيع خصائص الكينونات** : عن طريق التطبيع تجرى عملية تنقيح لنموذج الكينونات وقد يؤدى ذلك لمزيد من الكينونات إذا لم يحتو النموذج على الكينونات فى أبسط صورة لها ولنموذج الكينونات المطبع دور كبير فى إنتاج نموذج للبيانات مطبع بالتالى. ونموذج البيانات المطبع يصبح منظما بحيث أن :

- تحديث أى بيانات لن يكون إلا فى مكان واحد .
- حذف بيان لن يؤدى لفقدان بيانات أخرى بطريق الخطأ .
- كل خصيصة غير مفتاحية تمثل " حقيقة للخصيصة المفتاحية وتعتمد على كل الخصائص المفتاحية)، ولا تتعلق بغير الخصيصة المفتاحية .

وعند الوصول إلى تحقيق هذا الشرط يقال أن الكينونة على الصورة المطبعة الثالثة Third Normal Form (3NF) . ولكى نصل لهذه الدرجة من التطبيع يتم التطبيع على ثلاث درجات ، فنعدل فى الكينونات لتكون على الصورة المطبعة الأولى (INF) [لتحقق الجزء الأول من الشرط] ثم الثانية(2NF) [لتحقيق الجزء الثانى من الشرط] وأخيرا الجزء الثالث من الشرط].وتعطى مثالا لهذه العملية مطبقا على الكينونة " طلبية " ، وهى مصورة فى صورتها الأصلية فى الشكل (١٧) مع أمثلة لها .

ORDER

Order (order3, order data customer #, customer name [item #,item Quantity , item price] order

)rser #	Order data	Customer #	Customer_ name	Item#	Item quantity	Item Price
23	11/12/94	101	Mohamed	12	19	23.78
				14	2	145.99
				15	5	200.00
				23	1	96.00
126	11/12/94	102	Ahmed	11	2	67.89
				14	3	145.99
127	12/12/49	101	Mohamed	13	1	99.99
				14	2	145.99

شكل (17) الكينونة order فى صورتها الأصلية (قبل التطبيع)

الصورة المطبعة الأولى : تكـــــــون على الصورة المطبعة الأولى إذا لـــم تحتـو علـــى خصائص مكررة .

وعلى ذلك فإننا نضع الكينونة " طلبية " على الصورة الطبيعيـــة الأولـــى بحـذف الخصائص المكررة وهى رقم الصنف ، الكمية السعر . ويتم ذلك بإدخال نوع كينونة جديـــد يحمل الخصائص المكررة ولتسمية " تفصيل الطلبية **ORDER DETAIL** " . هذا النوع الجديد يرتبط بالنوع الأصلى " الطلبية " بعلاقة 1 : ن ، حيث أن كل طلبية قد يكـــون لـــها أكثر من كينونة تفصيلية لها ، بينما لا تتعلق الكينونة التفصيلية إلا بالطبية التى تفصلـــها . وبين شكل (١٨) التعديل الجديد .

ORDER

ORDER datail

ORDER (order #, order data, customer#, Customer name)
Order DETAIL (order # , item #, item Quantity , item price)
ORDER

ORDER #	ORDER data	Customer #	Customer name
123	11/12/94	101	Smith,s
126	11/12/94	102	HI-FI LTd
127	12/12/94	101	Smith,s

ORDER DETAIL

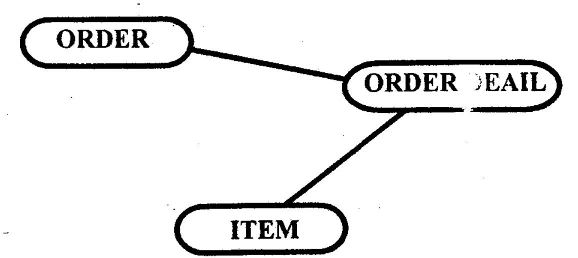
Order #	Item #	Item Quantity	, item price
123	12	19	23.78
123	14	2	145.99
123	15	5	200.00
123	23	1	96.00
126	11	2	67.89
126	14	3	145.99
127	13	1	99.99
127	14	2	154.99

شكل (١٨) الكينونة ORDER على الصورة المطبعة الأولى

الصورة المطبعة الثانية : تكون الكينونة على الصورة المطبعة الثانية إذا كانت كل خصيصة فيها تعتمد كل الخصائص المفتاحية [فى حالة تعدد الخصائص المفتاحية] .

ومثال ذلك فى الكينونة " تفصيل الطلبية " نجد أن الكمية " تعتمد على كل من الخصيصين المفتاحيتين ، رقم الطلبية ورقم الصنف ، فلا رقم الطلبية فقط ولا رقم الصنف فقط يمكن أن يدلنا على الكمية المطلوبة من صنف معين . أما " السعر " فيعتمد على " رقم الصنف " فقط رغم أن الكينونة لها خصيصتين مفتاحيتين ، وفى هذا إخلال بشرط الصورة الطبيعية الثانية .

ولتطبيع الكينونة فى هذه الحالة نخرج السعر من خصائص هذه الكينونة ، وتلحقة بكينونة أخرى ولتكن " الصنف " [لحسن الحظ هذه الكينونة موجودة أصلا ، ولكن لو لم تكن موجودة لأوجدناها بغرض التطبيع . ويبين شكل (١٩) ناتج هذه العملية .



ORDER(ORDER#,order data,customer#,Customer Name ORDER DETAIL (order#, item#, item quantity,item price ITEM (item#item pric)

RDER

Order #	Oeder data	Customer #	Customer name
123	11/12/94	101	Smih,s
126	11/12/94	102	HI-FI Ltd
127	12/12/94	101	Smith,s

ITEM

Item #	Item price
11	67.89
12	23.78
13	99.99
14	145.99
15	200.00
23	96.00

ORDER DETAIL

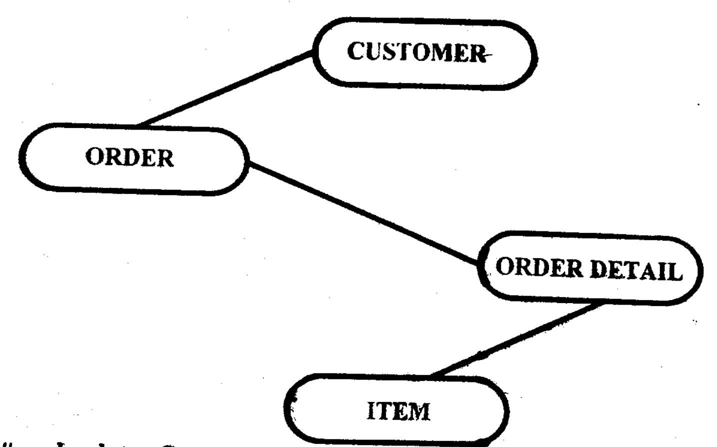
Oder #	Item #	Item quantity
123	12	19
123	14	2
123	15	5
123	23	1
126	11	2
126	11	3
127	13	1
127	14	2

شكل (١٩) كينونات على الصورة المطبعة الثانية

الصورة المطبعة الثالثة : تكون الكينونة على الصورة المطبعة الثالثة إذا لم تعتمد كل خصائص على الحقل المفتاحى .

فبالرجوع إلى الكينونة " طلبية " نجد أن خصيصة " إسم العميل " تعتمد على رقم العميل هذه العلاقة بين الخصائص غير المفتاحية هى ما يخل بشرط الصورة الطبيعية الثالثة.

ولتطبيع الكينونة لتلافة هذا العيب تخرج الخصيصتين وتلحقـــها بكينونـــة جديـــدة ولنسميها " العميل " [مرة أخرى هذه الكينونة موجودة أصلا ، ولـــو لـــم تكـــن موجـــودة لأوجدناها لهذا الغرض] ويبين شكل (٢٠) ناتج هذه العملية .



ORDER(Order#, order data , Customer#)
ORDER DETAIL(order#,item,item quantity)
ITEM (item#, item price)
Customer (customer #, Customer name)

ORDER

Order #	Order data	Customer #
123	11/12/94	101
126	11/12/94	102
127	12/12/94	101

ORDER DETAIL

ORDER #	Item #	Item quantity
123	12	19
123	14	2
123	15	5
123	23	1
126	11	2
126	14	3
127	13	1
127	14	2

ITEM

Item #	Item price
11	67.89
12	23.78
13	99.99
14	145.99
15	200.00
23	96.00

COUSTMER

Customer #	Customer name
101	Mohamed
102	Ahmed

شكل (٢٠) كينونات على الصورة المطبعة الثالثة

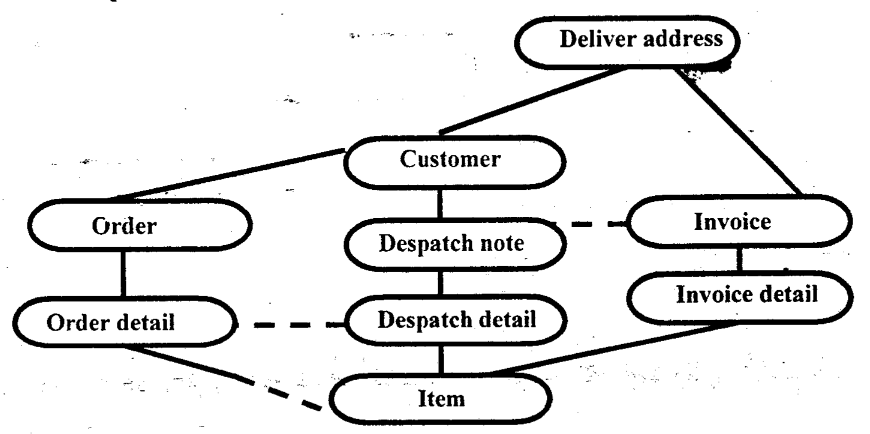
وخلاصة القول أن تطبيع أنواع الكينونات يؤدى إلى نموذج للبيانات يشكل أساسا لتصميم جيد لقاعدة البيانات لأنها :

تحلل أنواع الكينونات صورها .

تضمن عدم تكرار البيانات .

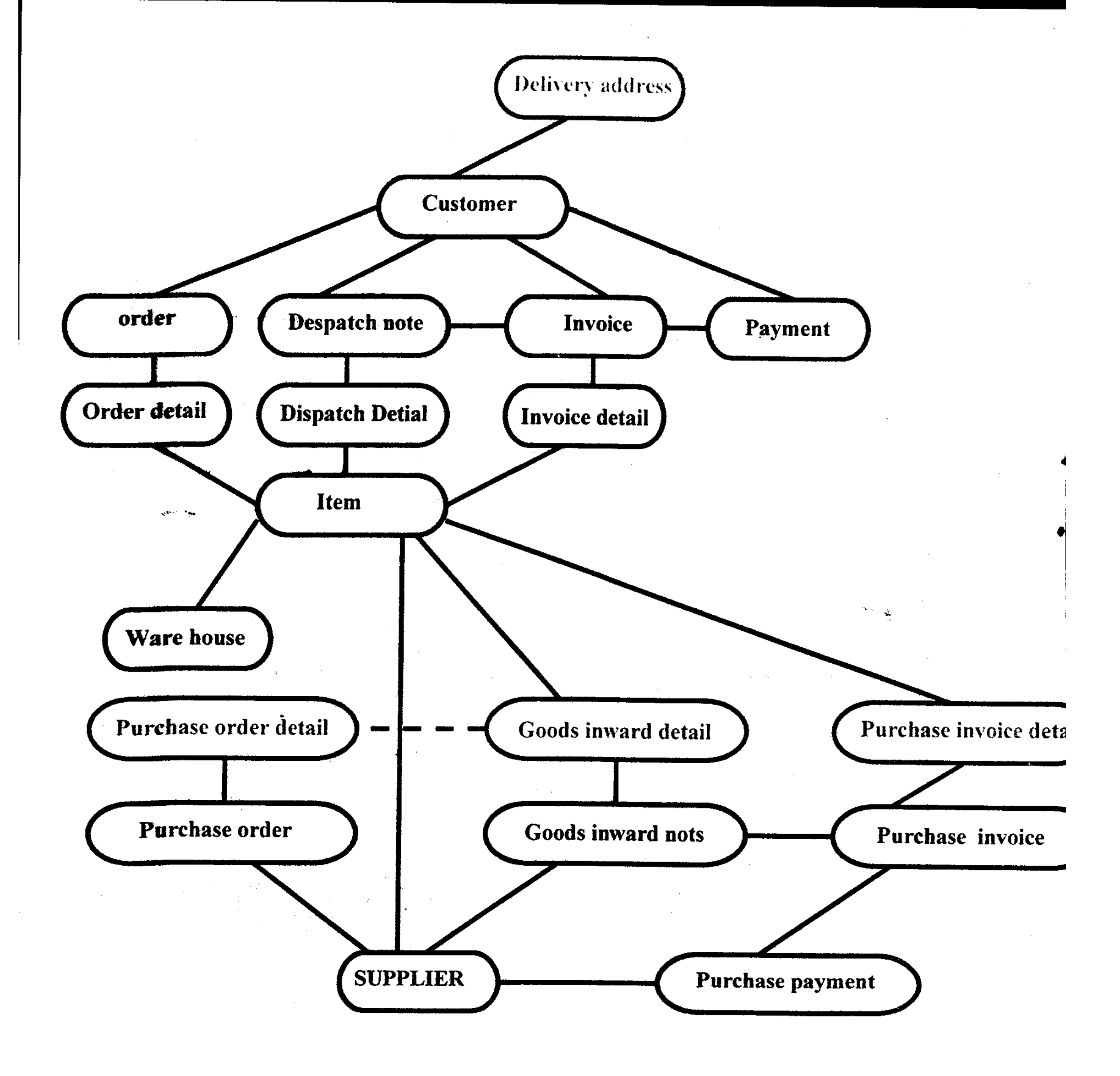
تسمح باستقلال بيانات الكينونات وعدم إعتمادها على كينونات أخرى .

وبعد الإنتهاء من تطبيع كافة أنواع كينونات هذه المنطقة يتم دمجها فى نموذج موحد كما هو موضح بشكل (٢١) موضح فيه تطبيع كينونة " طلبية " بإضافة كينونة " تفصيل الطلبية " كذلك تطبيع كينونة " العميل " بإضافة كينونة " عنوان التسليم DELIVERY ADDRES " حيث أنه من الخصائص المكررة " العميل "] .



شكل (٢١) مخطط كينونات عملية معالجة الطلبيات بعد التطبيع

(٧) **دمج النماذج :** يبين شكل (٢٢) الصورة النهائية لنموذج الكينونات والعلاقات بينها لكل مناطق الشركة التى تناولها التحليل وهى : معالجة الطلبيات ، مراقبة المخزون ومعالجة المشتريات ، وبعد إجراء التطبيع على كافة الكينونات.



شكل (٢٢) "مخطط الكينونات الكامل"

(٨) **إختبار الملاءمة** : يتم إستخلاص نموذج البيانات مستقلا عن العمليات التى تحتاج إلـــى تلك البيانات ، وذلك تمهيدا لوضع قاعدة بيانات تفى بأية مهمة تتطلبها المنشأة فى الحـــاضر أو المستقبل وللتأكد من هذا يجرى التحليل الوظيفى **Functional analysis** الذى يبيـــن

كيف تطبق العمليات على نموذج البيانات ، ومن ثم فهو يعمل كإختبار لمدى ملاءمته للعمـــل .ولكل عملية من العمليات يقدم التحليل الوظيفى الذى يتسم تطبيقه على نموذج البيانات مـــا يلى .

- قائمة بأنواع الكينونات التى يجب الوصول إليها .
- ترتيب الوصول لتلك الكينونات .
- المعيار المتبع لإختيار الكينونات ، فقد يتم إختيار كينونة " طلبية " مـــن خـــلال رقــــم الطلبية أو قد يتم إختيار كافة الطلبيات ذات علاقة معينة بكينونـــة " عميل " .
- طبيعة الوصول (إسترجاع ، تحديث ، محو ... إلخ) .

يتم بعد ذلك رسم المخطط الوظيفى ، وهو مخطط للفئـــات الفرعيـــة مـــن نمـــوذج البيانات . وهو يبين الاتجاهات والتفريغات التى تتم داخل النموذج للوصول إلـــى الكينونـــات المطلوبة لتنفيذ العملية كما يتم تكوين جدول ، ضم المعايير والنتائج المتحققة . وتعطى لذلك مثالا تطبيقيا من شركة "أبو الفتوح" .

الوظيفة : عمل قائمة بالأصناف التى طلبها العميل منذ أربعة أيام ولم تنفذ حتـــى تاريخـــه ، وتضم القائمة (رقم الطلبية ، تاريخ الطلبية ، رقم الصنف ، بيان الصنف ، السعر ، الكمية) .

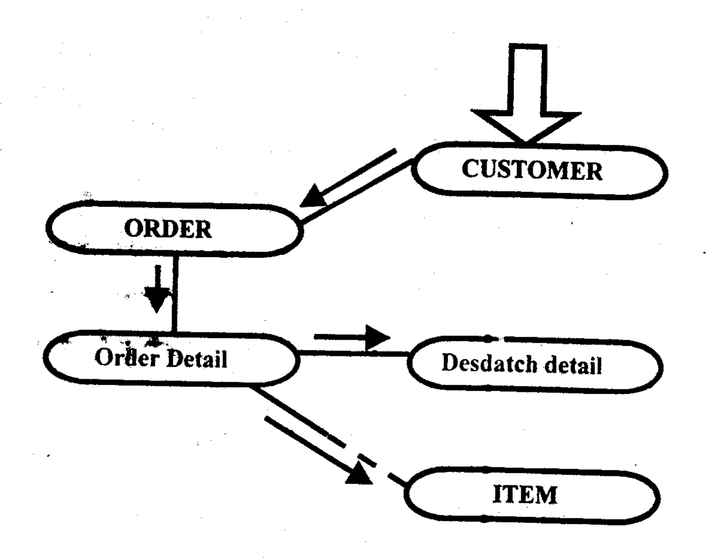
يجب التنويه بأنه ليس المقصود من التحليل الوظيفى هو تنفيذ الوظـــــائف فعـــلا ، فهذه مهمة البرنامج التطبيقى الذى سيوضح لهذا الغرض ، ولكن المقصـــود هـــو التأكد من إمكانية تنفيذها وبالنسبة للوظيفة الأولى يجب عمل الخطوات التالية :

(١) إسترجع الكينونة " عميل " التى يتم إختيارها بمعرفة " رقم العميل "

(٢) إسترجع جميع الطلبيات التى تخص ذلك العميل من خـــلال علاقـــة " عميـــل / طلبية " بحيث يرجع تاريخ البحث إلى أربعة أيام مضت .

(٣) إسترجع تفاصيل تلك الطلبيات من خلال العلاقة " طلبية / تفاصيل الطلبية " .

(٤) إسترجع تفاصيل الصنف من خلال العلاقة " تفاصيل الطلبية / الصنف " .



Entity accessed	Relationship	Selection Criterion	Action
Customer	-	Customer # = ?	Retrieve
Order	Customer/Order	Order data < today –4	Retrieve
Order Detail	Order/Order detail / Dispatch Detail	All	Retrieve
Item	ORDER Detail / item	All	retrieve

شكل (٢٣) المخطط الوظيفى للكينونات ، الوظيفة رقم ١

قاموس البيانات

قاموس البيانات يعتبر معجم مثله مثل المعاجم اللغوية والقواميس وإن إختلفت دلالته قليلاً ، فهو ليس معجم لغوى يوصل بين لغتين ، لكنه مجمع يضم التعريف والتوصيف لكل البيانات والعمليات والإجراءات والمخازن والسياسات المستخدمة فى نظام المعلومات.

وقاموس البيانات فى حد ذاته ليس ذا فائدة على الإطلاق إن لم يشمل كل ما يتعلق بالنظام محدداً بدقة شديدة سيان كان آلياً أويدويا ، وبفضله تتوجد المفاهيم والتفاصيل والدلالات بين مختلف العاملين فى النظام ، مما لا يدع لمحلل النظم مجالاً للإجتهاد أو تفسير عنصر بيان أو إجراء وفق ما يراه من وجهة نظره .

ويمكن تحديد أهداف قاموس البيانات على النحو التالى :

١- إنشاء موسوعة كاملة عن كل عناصر النظام .

٢- إتاحة تعريفات قياسية موحده عن كل عناصر النظام .

٣- تحديد وتوصيف الكيانات الخارجية المتعاملة مع النظام .

٤- إتاحة مداخل متعددة وفهرسة متعامدة .

٥- تحديد الوحدات المتكاملة فى البرامج والسياسات MODULES.

٦- حسم مشكلة المتشابهات والمترادفات من التعريفات .

٧- تبسيط عملية تعديل أى عنصر فى النظام وجعلها مركزية مما يضبط ويوثق عملية التغيير .

٨- يساعد على تحديث وصيانة النظام بعد إنشاءه .

٩- يساند عملية تصميم النظام .

١٠- يعتبر جزء مكمل ورئيسى وحيوى وضرورى لخرائط التدفق .

إذا بدونه تصبح خرائط التدفق محدودة الجـــدوى وتتحـول إلـى مجـرد أشـكال معقدة غير مفهومة ، حقيقة قد تعطى الخرائط فكـــرة شـاملة عـن النظـام لكـن دقـائق وخفايا النظـــام تكمـن فـى قـاموس البيانـات ، أى أنـهما شـيئا واحـدا وتوأمـان لا ينفصلان ولاغنى عنهما للآخر ، إذا بدون خرائط التدفـق فـلا معنـى ولا مـبرر لإنشـاء قاموس البيانات أو قـاموس النظـام ويتحـول إلـى مجـرد نصـوص لا تفيـد دارس أو باحث ولن يلجئ إليه إنســان .

العلاقة بين قاموس البيانات وخرائط التدفق :

تنشأ هذه الرابطة التى أشرنا إليها عبر أحـــد منطلقيـن :

الأول : وجود مدخل وحيد لكل بيان ورد فى خرائط التدفــق ، آيـا كـان مسـتوى تجريـد الخريطة وعلى إمتداد النظام ، كما آن هناك مدخـــل وحيـد لكـل مخـزن بيانـات ورد فى خريطة التدفــق.

الثانى : توجد مداخل أخرى لكل مالم يـرد فــى خرائـط التدفـق سـيان كـانت بيانـات يراها محلل النظام ضروريـــة لإعطـاء النظـام التكامليـة المطلوبـة ، آو إضفـاء بعض البيانات المساعدة التى يلجأ إليها لنفــس الغايـة والـهدف .

أنواع قواميس البيانات :

يمكن تقسيم قواميس البيانات إلى ثلاثـــة أنـواع هــى اليـدوى – النصـف آلــى – الآلى ، ويتوقف إختيار نوع القاموس على عوامـل عـدة أبرزهـا حجـم النظـام وزمـن الإستجابة المطلوبة ومدى إعتمادية النظام علــى القـاموس ، ولانغفـل تكلفـة القـاموس أيضا كأحد المؤثرات فى إختيـــاره .

١ – القاموس اليدوى :

وهو عبارة عن مجموعة مرتبة مـــن البطاقـات الورقيـة ، وأحيانـا تسـتخدم الدفاتر والكراسات فى إعداده . مثل هذا النـــوع يحتـاج جـهد كبـير فـى إعـداده . ويعتبر القاموس اليدوى من أفضل أنواع القواميـــس فــى حـالات التحليـل الـهيكلى من القمة إلى القاع لأنه يمتازبعدة مزايـــا أبرزهـا :

أ- التوصيف الدقيق لعناصر النظام الأربع [البيانات - التدفق - الملفات - المعالجات] .

ب- البساطة الشديدة فى الإنشاء والصيانة والتحديث .

حـ- إمكانية خلق الفهرسة الإحداثية ، إن إشترك فى إعداد القاموس أحد خريجى قسم الوثائق والمكتبات .

د- يخصص كارت أو بطاقة لكل عنصر يتم توصيفه .

هـ- يمكن ميكنة القاموس اليدوى إذا خزنت المعلومات الواردة به على حاسب شخصى مما يساعد على عمليات الفرز والدمج والإضافة والتعديل .

و- غير مكلف مادياً .

٢- القاموس الآلى :

يقصد بلفظ الآلية هنا ، إستخدام بعض البرامج الجاهزة فى إنشاء المعجم ويتم إعدادها " توضيبها " بواسطة محلل النظام حتى تتمشى مع الهدف المنشود.

والمعجم الآلى يحقق مايلى :

أ- يقبل توصيف العناصر الأربعة للنظام .

ب- يرفض أى تكرارات .

ج- يسمح بالتحديث والتعديل والحذف والإضافة .

د- يجرى التطابق بين الأسماء والمتشابهات والمترادفات .

هـ- يقدم قوائم مطبوعة منظمة ذات هيئة Format موحدة يسهل الإطلاع عليها

و- يرد الفهارس المتقاطعة ويقدم قوائم بالعناصر TERMS أو الحدود التى لم تعرف أو توصف .

ح- بسيط التركيب ويستطيع أى محلل متمرس التعامل معه .

ط- مكلف نسبياً .

٣- القاموس النصف آلى [المختلط] :

هو عبارة عن مزج بين القاموس اليدوى والقاموس الآلى .

مكونات قاموس البيانات :

١- عنصر البيانات :

عبارة عن وحـــدة بيـــان لا يمكـــن تقســـيمها حتـــى لا تفقـــد دلالتـــها المنطقية ، فالتاريخ عبارة عـــن يـــوم / شـــهر / ســـنة ، ومـــع ذلـــك فإنـــه عنصر بيان أو دعنا نطلق عليه بيـــان ذرى [أى لاينقســـم فـــى عـــرف نظـــم المعلومـــات] .

۞ **عناصر مكونات البيانات :** ويوضع ضمنها وكلما وجـــد مـــا يلـــى :

أ- الأســـماء المســـتعارة ALIASES .

ب- البيانـــات المرتبطـــة RELATED .

ج-المدى والقيمـــة .

د- الطول بوحدة البـــايت .

هـ- الكود أو الرموز المستخدمة لمقابلـــة بيانـــات .

و-أساليب التحقق والمراجعـــة .

ن- حجم البيانـــات (ســـنويا - شـــهريا - يوميـــا) والحجـــم الأقصـــى والحجـــم الأدنـــى .

ح- أساليب التأمين وإختيـــار الصلاحيـــة .

ط- الأســـبقية .

ك-البيانات المخلقة وغير مدخـــلات النظـــام .

ل- توصيف النماذج فيزيائيـــا ومنطقـــى .

مثال:

عنصر البيان المسمى " التاريخ " يظهر فى أمر الشـــــراء [redacted] ســداد شـيك وحتى يتحول إلى عنصر بيان فريد UNIQUE يجب إعطـــاءه مســـميات مختلفـــة مثـــل :

[تاريخ الســـداد].

[تاريخ إصدار الشـــيك]

[تاريخ طلب الأصنـــاف]

الأسماء المستعارة :

تنشأ ظاهرة الأسماء المستعارة فى العديد والكثير والمتنوع من نظم المعلومات نتيجة إستخدام مسميات متنوعه لبيان واحد ، كالقول ، الحاسب الإلكترونى الحاسوب ، المخ الإلكترونى ، الكمبيوتر مثلما تنشأ من عدم توحيد أسماء ، المتغيرات VARIABLE NAMES بين معدى برامج النظام ، ويفترض فى قاموس البيانات إحتوائه على تلك الأسماء المستعارة والمتشابهات ليس فقط تحت المسمى الصلى ، لكن من خلال مداخل مستقلة لكل مسمى .

حاسب الكترونى ـــ جهاز يعالج البيانات إنظر كذلك .

[الحاسوب] [الكمبيوتر]

البيانات ذات العلاقة :

وهى ليست أسماء مستعارة ، بل بيانات لها علاقة بالبيان الأصلى ولم بوصفها التاريخ ١٩٣٩/١/٤ يمكن آن يكتب هكذا ٤ يناير ١٩٣٩ ويكتب كذلك ٤-١-١٩٣٩ كما يكتب ١٩٣٩٠١٠٤ .

القيمة والمدى :

يقصد بالمدى قيمة ثابته لا تتغير على إمتداد النظام مثل الأكواد لبعض الأقسام .

الكود	القسم
٧٦	المبيعات
٥٠	المكتبة
١	المشتريات
٢	مجلس الإدارة

ويقصد بها أيضا قيمة متغيرة مثل الرصيد – سن العامل فى مدى من ١٨-٦٠ سنة .

عنصر بيان (الإسم)	* الرقم
إسم المتغير المناظر	GOV – CODE ....
التعريف عبارة عن ثلاثة أرقام يسمح فيها بإستخدام الصفر	
التقسيم والأكواد	٠٠١ القاهرة ٠٠٢ الإسكندرية ٠٥٥ الشرقية ...... إلخ .
الهيئة أو الفورمة	
الإستخدام	سجل مواطن
قواعد التخليق	لاتوجد
الوحدات	لا توجد
طول الحقل	٣ بايت
أسماء مستعارة	لا توجد
قواعد المراجعة	الرقم الأول من اليسار لابد أن يساوي صفر
المدى	مدى
العلاقة مع البيانات	العنوان – محل الميلاد – محل العمل / هل هناك كرت آخر: نعم لا

شكل (٢٤)

يوضح عنصر البيانات

* يقصد بالرقم المسلسل لصفحة القاموس

ونلاحظ من هذا النموذج مــا يلـى :-

أ- يجب أن يكون إسم عنصر البيان ذاتى ومعــبر وســهل الإســتخدام والتذكــر وليــس مختصر أو ممتــد .

ب-التعريف ، وهو شرح لإسم البيان وإلــى أى نــوع ينتمــى مثــل هجــائى – رقمــى مختلط – تاريخ منطقى ، وكيف يتولد هذا العنصر فــى النظــام المعلومــات .

حـ- يجب النص على الوحدات ، ففى عنصــر بيــان الــوزن ينــص صراحــة علــى وحدة القيــاس .

الهيئــة أو الفورمــة FORMAT :

وتستخدم فى هياكل البيانات ، أو مجموعات البيانات ، ففــى التــاريخ هــل يذكــر علــى هيئة .

يوم/ شهر / سنة	YY/MM/DD
أو يذكر على هيئة	YY/DD/MM

قواعد القبــول :

هناك العديد من عنــاصر البيانــات ذات قيــم يجــب إختبارهــا ، إســم الشــخص لايجب إحتوائه علــى أرقــام ، كمــا آن الــوزن يجــب أن يكــون موجــب ، ودرجــة الحرارة تحتمل آن يكون موجبة أو سالبة ، كما آن ســن مــن يعمــل فــى الحكومــة لا يزيد عن ٦٠ عاما إلا فى مصر ففيها العجب الكــل يمــد لــه وكأنــه قديــس مخلــد لا يمكن الإستغناء عنه ، ولم تنجب الإمــهات مثلــه .

فهرس عناصر البيانات :

فى بعض المعلومات عندمــا تتعــاظم كميــة عنــاصر البيانــات فمــن الأفضــل إعــداد فهرس خاص بها يسهل التعامل معه ويكون مدخلا لعناصر البيانـــات المختلفــة فــى النظــام .

٢ - هيكل بيان :

وتعنى عناصر بيانات مرتبطة ببعضها البعض ، فالعنوان هيكل بيان يتركب من رقم المنزل – إسم الشارع – الحى – المدينة – المحافظة – الدولة .

هيكل بيان		الرقم المسلسل
الإسم	أمر شراء	
التعريف	أمر شراء يرسله العميل إلى شركة توزيع السلع	
عناصر هيكل	أرقام خرائط التدفق	الشروط
رقم العميل الإسم رقم التليفون العنوان		

شكل (٢٥)

٣-تدفق بيان :

هو هيكل أو مجموعة هياكل بيانات تتحرك من مكان لآخر ومادام لها منبع فإن لها مصب قد يكون مخزن بيانات أو مرحلة معالجة أو التوجه إلى كيان .

	الرقم	تدفق (حركة) ← حركة بيانات →
حركة بيانات	تجميع أوامر التوريد	
رقم المنبع		
رقم المصب		
التفصيلات :	حصر الأصناف المطلوبة وحصر الأصناف لكل عميل	
هياكل البيانات أمر شراء بيانات العميل بيانات الأصناف	أقصى طلب للأصناف ٣٣٠ طلب / يوم حد طلب الصنف فى المخزن تغطى فى حدود ٢٥٠ يوما	

شكل (٢٦)

٤- مخازن البيانات :

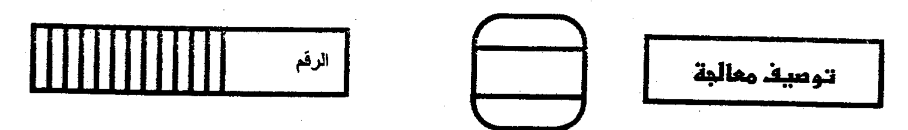
وهو بيانات فى حالة سكون ، وقد يكون سكون مؤقت فى إنتظار التحرك أو السكون دائم كما فى الآرشيفات التاريخية .

مخزن بيانات	الرقم	
إسم المخزن :	العملاء	الرقم :
الوصف :	تجميع البيانات الخاصة بـ	
المدخلات	المخرجات	المحتمه
xxxxxxxx	xxxxxxxxx	xxxxxxxxx
xxxxxxxx	xxxxxxxxx	xxxxxxxxx
وصف المخزن : دفتر أستاذ مدون يدويا ويتم التحديث يدويا كل يوم		

الشكل (٢٧)

٥- العمليات : (توصيف معالجة بيانات) :

حيث يتم توصيف معالجة البيانات بإستخدام عدة طرق الطريقة (HIPO) Hierar Chical Input – Process – Output أو بإستخدام شجرة القرارات . والشكل التالى يوضح نموذج لتوصيف معالجة بيانات .



الوصف :		
المدخلات	**المعالجة**	**المخرجات**
xxxxxxxx	xxxxxxxxxxxxxxx	xxxx
xxxxxxxx	xxxxxxxxxxxxxxx	xxxx
xxxxxx	xxxxxxxxx	xxxxx
التنفيذ الحالى : يتم يدويا بالرجوع إلى قائمة العملاء .		
مكان المعالجة : قسم إستقبال xxxx		

الشكل (٢٨)

٦- توصيف كيانات النظام :

يجب تحديد الكيانات الخارجية المتعاملة مع النظام وفق حالة كل كيان فإذا كان الكيان شخص يشمل التوصيف الآتى :

أ- الإسم كما هو مدرج فى خرائط التدفق .

ب-حركة البيانات المرتبطة بالكيان .

ج- عدد الأشخاص فى اليوم .

د- معدلات تزايد العدد .

هـ- الأنواع والمستويات .

و- أهم وإبرز الأشخاص .

أما إذا كان الكيان نظام معلومات آخر فيجب أن يوصف على النحو :

أ- لغات البرمجة المستخدمة .

ب- وصف الكيان الآلى وعناصره .

حـ- نظام التشغيل .

د- إمكانية الربط بين النظامين .

متى يستخدم قاموس البيانات :

متى تم إنشاء قاموس البيانات فإنه يستخدم فى الحالات الثلاث الآتية :

١- مرحلة تحليل النظام .

٢- مرحلة تصميم النظام .

٣- مرحلة المراجعة وضبط الأداء .

الفصل الخامس

التحكم فى عرض البيانات باستخدام لغة الاستعلام SQL

الفصل الخامس

التحكم فى عرض البيانات باستخدام لغة الاستعلام SQL

تعد نظم قواعد البيانات العلاقية من الأنظمة التى ترتكز على إستخدام لغة SQL، وكلمة SQL أصلها SEQUEL وهو الإسم الذى أطلقه معمل التطوير والأبحاث بشركة IBM على اللغة الجديدة التى تم تطويرها لإستخدامها فى تصميم نظم إدارة قواعد البيانات . وكلمة (SQL) إختصار للعبارة Structured Query Language ، أى لغة البحث والإستفسار المركبة لأنها لغة بحث وفى نفس الوقت لغة لبرمجة قواعد البيانات . وتستخدم لغة SQL لتمثيل قواعد البيانات العلاقية زتتيح للمستخدم (سواء كان مستخدم نهاية طرفية أو مبرمج تطبيقات) إمكانيات كبيرة ومرونة فائقة لتشغيل البيانات .

ويلاحظ من الشكل السابق أن المستخدمين على إختلاف أنواعهم يقومون بتشغيل نفس البيانات فى نفس الوقت حيث يمكن التحكم فى عمليات التشغيل بحيث يتم جماية كل مستخدم من التأثير على المستخدم الآخر .

ويتيح نظام الـ SQL التعامل مع نوعين من الجداول : جدول القاعدة Base Tables وهى حقيقة والأشكال أو الصور Views وهى جداول مخلقة من الجداول الأصلية وبصور مختلفة .

وتتيح لغة (SQL) للمستخدم التعامل مع أكثر من جدول لقاعدة (Base Table) أو صورة مثل (VSAM File) فى نفس الوقت .

وتحتوى لغة (SQL) على جزئين رئيسيين ، جزء خاص بتعريف البيانات Data Definition Language (DDL) وجزء خاص بتشغيل البيانات Data Mainpulation Language(DML) .

٢) المكونات الرئيسية للغة SQL :

يتكون نظام SQL من أربعة أجزاء رئيسية تمثل المكونات الأساسية لتلك اللغة وهى :

أ- جزء ما قبل برنامج الترجمة Precompiler :

وهذا الجزء عبارة عن معالج Processor لبرامج التطبيق التى تحتوى على جمل SQL . ويقوم بتجميع هذه الجمل داخل نموذج قاعدة بيانات جاهز للطلب Datebase Request Module أو (DBRM) ثم إستبدالها فى البرنامج الأصلى بمجموعة نداءات CALLS للجزء المراقب لزمن التشغيل .

ب- جزء الربط Bind :

يؤدى هذا الجزء دور الترجمة لواحد أو أكثر من DBRM للحصول على كود الآلة Machine Code الخاص بها بما فيها الندءات لمدير البيانات المخزونة Stored Date Manager.

ج- جزء مراقب زمن التشغيل Runtime Supervisor:

ويقوم هذا الجزء بمراقبة تنفيذ برنامج SQL إثناء التنفيذ . فمثلاً عند طلب بعض عمليات قواعد البيانات من قبل برنامج معين فإن وحدة التحكم Controller توجه هذا الطلب أولاً إلى مراقب زمن التشغيل وذلك عن طريق النداءات التى يضيفها ما قبل برنامج الترجمة Precompiler. ثم من مراقب زمن التشغيل إلى خطة التطبيق Application Plan والتى توجه مدير البيانات المخزونة لتنفيذ العملية المطلوبة .

د-مدير البيانات المخزونة Stored Data Manager:

يقوم مدير البيانات المخزونة بإدارة قاعدة البيانات الفعلية بالإضافة إلى تخزين وإسترجاع السجلات حسب الحاجة . والشكل التالى يوضح ذلك .

مكونات جمل SQL:

تتكون لغة SQL من سلسلة من الجمل التـى تـؤدى مـهاماً معينـة ، وذلـك علـى النحو الموضح فى جدول (١) . إذ أنه هناك جمـل تقـوم بدورهـا بإنشـاء قواعـد البيانـات ، وجمل تقوم بإضافـة وحـذف الصفـوف فـى جـدول ، وأخـرى تعمـل علـى إسـترجاع الصفوف من جدول أو مجموعة مـن الجـدول . بالإضافـة إلـى ذلـك هنـاك جمـل أخـرى تتعامل مع تأمين البيانات إلى جـانب تكاملـها . ويتـم إنشـاء هـذه الجمـل وفقـاً لقواعـد معقدة تختلف نوعاً ما من نظام قواعد بيانات إلـى آخـر . لكـن بالنسـبة للمسـتخدمين ، لا تمثل هذه الإختلافات ثمة مشـكلة .

جدول (١)

بعض جمل SQL الشائعة

إسم الجملة	الوصف
Create Index	إنشاء فهرس بالنسبة مجموعة من الأعمدة فى الجدول .
Create Table	إنشاء جدول خالى فى قاعدة البيانات
Create View	إنشاء طريقة عرض
Delete	حذف صفوف من الجدول
Drop Index	حذف فهرس من قاعدة البيانات
Drop Table	حذف جدول من قاعدة البيانات
Drop View	إزالة طريقة عرض من قاعدة بيانات
Insert	إضافة صفوف إلى جدول
Select	إسترجاع صفوف من جدول
Update	تغيير قيم البيانات لعمود أو أكثر فى الجدول

ويطلـق علـى جمـل Create Table و Drop Table و Create View و Drop View و Create Index و Drop Index جمـل Data Definition Language (DDL) ،

بينما يطلق على جمل Insert و Delete و Update و Select جمل Data Manipulation Language (DML) . وفى معظم نظم قواعد بيانات وخاصة فى هذه الآونة ، نادراً ما يتم تنفيذ جمل DLL عندما ترغب فى إنشاء بنية قاعدة البيانات . بل أنك بدلاً من ذلك ستستعين بأداة مرفقة بنظام قاعدة البيانات تسمح لك بإمكانية ملء جميع المعلومات داخل جدول ، أو إنك ستستخدم معالجاً سوف يساعدك على إنشاء الجدول أو الفهرس .

وعادة ما يضم قاعدة البيانات سمة تسمح لك بإمكانية إنشاء جمل SQL من خلال التعريفات التى قمت بإدخالها . ثم قد تستخدم بعدئذ جمل SQL تلك لإنشاء نسخة من قاعدة البيانات على كمبيوتر آخر أو لوضعها فى التطبيق الخاص بك إذا كنت ترغب أن يكون المستخدمون قادرين على إنشاء بنيات قواعد بيانات بصورة سريعة .

أنواع بيانات SQL:

ينبغى أن يوجد لدى كل عمود فى الجدول نوع بيانات مرتبط به . ولابد أن يتوافق نوع البيانات الذى نختاره للعمود مع نوع البيانات الخاص بمتغيرات Visual Basic ويوضح جدول (٢) بعضاً من أكثر أنواع البيانات المستخدمة شيوعاً من خلال لغة SQL ، فضلاً عن أنواع البيانات المعادلة لها فى Visual Basic .

جدول (٢)

بعض أنواع بيانات SQL الشائعة

نوع بيانات SQL	نوع بيانات Visual Basic	الوصـــــــف
Char	String	سلسلة أحرف ذات طول ثابت
Date	Date	قيمة تحتوى على قيمة تاريخ وزمـــن (متاحـــة مـــن خـــلال Oracle فقط)
Datetime	Date	قيمة تحتوى على قيمة تاريخ وزمن (متاحة من خـــلال Sql Server فحسب)
Decimal	Curren	القيمة الرقمية الدقيقة للحجم المحدد
Float	Double	Floating Point number مـــكون من 64 بت .
Int	Long	عدد صحيح مكون من 32 بت.
Mony	Currency	قيمة رقمية دقيقة (متاحة من خلال SQL Server فقط) .
Number	Currency	قيمة رقمية دقيقة (متاحة من خلال Oracle فقط)
Real	Single	Floating – Point number مكون من 32 بت .
Smallint	Integer	عدد صحيح مكون من 16 بت .
Varchar	String	سلسلة أحرف ذات طول متغير .

تنقسم أنـواع البيانـات إلى أربعة أنـواع رئيسية : القيـــم الرقميـــة الدقيقـــة Floating Point وقيم سلاسل الأحرف وقيم البيانـــات . بالإضافـــة إلـــى ذلـــك ، تقـــدم معظـــم وحدات الخدمة كذلك العديد من أنواع الأخرى التى يمكنـــك الإختيـــار منـــها .

أنـواع البيانـات الرقميـة الدقيقـة :

تمثل أنواع البيانات الرقمية الدقيقـــة أرقامـــاً مـــن خـــلال إســـتخدام قيمـــة دقيقـــة . وتنقسم هذه الأنواع من البيانات إلـــى قســـمين فرعييـــن : قيـــم الأعـــداد الصحيحـــة والقيـــم

العشرية المضغوطة . وتقوم قيم الأعداد الصحيحة بتخزين الأرقام الخاصة بها بمثابة قيمة ثنائية ويتيح ذلك المزيد من إمكانية التخزين الفعال بالنسبة للأرقام الكبيرة أكثر مما هو الحال عندما تستخدم قيمة عشرية مضغوطة .

من ناحية أخرى ، تقوم القيم العشرية المضغوطة بتخزين الأرقام بمثابة سلسلة أعداد رقمية (من 0 إلى 9) . ويتم إستغلال 4 بت لتمثيل قيمة من Zero إلى 9 . وتسمح لك معظم وحدات الخدمة الخاصة بقواعد البيانات بإمكانية تحديد عدد الأعداد (من 0 إلى 9) المطلوب عندما تقوم بتحديد نوع البيانات .

وتتسم القيم الرقمية بأنه عندما يتم إجراء العملية الحسابية من خلالها فإنك لن تفقد عنصر الدقة على وجه الإطلاق .

أنواع بيانات Floating Point :

تمثل أنواع بيانات Floating Point أرقاماً بحيث يتم تقسيمها إلى جزئين : الجزء العشرى من اللوغاريتم والأسس . ويتم التعبير عن Floating Point من خلال ذكر قيمة مضروبة فى 10 مرفوعة لقدرة ما ، على سبيل المثال ، يتم كتابة قيمة 12.345 بمثابة 5 10× 1.234 وغالباً ما يتم عرضها على الكمبيوتر بشكل 1.2345E5 حيث تشير E إلى 10 مرفوعة إلى هذه القدرة . وفى هذا المثال ، يتمثل الجزء العشرى من اللوغاريتم فى 1.2345 والأس فى 5 .

ونظراً للأسلوب الذى يتم من خلاله تخزين Floating Point Numbers فإنها تعتبر دقيقة فحسب بالنسبة للعديد من الأماكن العشرية .ويسمح لك بإمكانية تمثيل أعداد كبيرة للغاية من خلال مساحة أقل مما هو مطلوب فى حالة ما إذا قمت بتخزين كل عدد على حدة . وبوجه عام ، تتسم القيم الفردية بالدقة بالنسبة لحوالى خمسة أماكن عشرية أما القيم الزوجية فهى تتسم بالدقة بالنسبة لحوالى عشر أماكن عشرية .

أنواع بيانات سلاسل الأحرف :

تحتفظ أنواع بيانات سلاسل الأحـــرف بالمعلومـــات الخاصـــة بـــالحروف . إذ يوجـــد نوعان من سلاسل الأحرف : سلاسل الأحـــرف ذات الطـــول الثـــابت وسلاســـل الأحـــرف ذات الطول المتغير. أما بالنســـبة لسلاســـل الأحـــرف ذات الطـــول الثـــابت فتحتفـــظ دومـــاً بنفـــس مقدار المساحة داخل الجدول ســـواءً قمـــت بتخزيـــن حـــرف واحـــد فـــى العمـــود أو قمـــت بتخزين خمســـين حرفـــاً . علـــى الجـــانب الآخـــر ، تحتفـــظ سلاســـل الأحـــرف ذات الطـــول المتغير بالحروف التى توجد لدى سلسلة الأحـــرف فحســـب فضـــلاً عـــن بعـــض المعلومـــات الأخرى بصدد الإحتفاظ بطول سلســـلة الأحـــرف .

وبوجة عام ، ينبغـــى عليـــك إختيـــار سلاســـل الأحـــرف ذات الطـــول المتغـــير عـــن سلاسل الأحرف ذات الطول الثابت . إذ يعنى ذلك توفـــير قـــدر مـــن المســـاحة المتاحـــة فـــى قاعدة البيانات لاسيما إذا كان مقدار البيانـــات الـــذى تقـــوم بتخزينـــه فـــى العمـــود يختلـــف بشكل كبير من صف إلى آخر . بالإضافة إلـــى ذلـــك يتيـــح لـــك إســـتخدام سلاســـل الأحـــرف ذات الطول المتغير كذلـــك إمكانيـــة إنشـــاء سلاســـل الأحـــرف التـــى تحتـــوى علـــى الحجـــم الأقصى بصورة أكبر ويعد ذلك مفيداً فى الحالات التى قد يكـــون لديـــك فيـــها قيمـــة ضخمـــة غير عادية لا ترغب فى إقتضابها ، وذلك مثـــل إســـم الشـــخص .

من ناحية أخرى ، تعتبر سلاسل الأحـــرف ذات الطـــول الثـــابت جيـــدة حينمـــا يظـــل حجم كل قيمة ثابتاً بصورة نسبيه ، وذلك كما هـــو الحـــال بالنســـبة لإختصـــار إســـم الولايـــة المكون من حرفين أو كـــود أداة التعريـــف الخاصـــة بـــالمنتج . ويعـــد ذلـــك الأمـــر حقيقـــاً لاسيما فيما يتعلق بسلاسل الأحرف الصغيرة حيـــث يشـــغل الجـــزء العلـــوى الإضـــافى الـــذى يعنى بمهمة متابعة الطول الحقيقـــى لسلســـلة الأحـــرف حـــيزاً أكـــبر مـــن مســـاحة سلســـلة الأحرف ذاتـــها .

أنواع البيانات الخاصة بالتاريخ :

تتسم أنواع البيانات الخاصـــة بالتـــاريخ بأنـــها ممـــيزة لكـــل نظـــام قواعـــد بيانـــات معين ورغم عدم وجود توافق بصورة كبيرة بيـــن التطبيقـــات الخاصـــة بالشـــركات المنتجـــة لقواعد البيانات إلا أن اللجوء إلى حلول بديلة يعـــد أمـــراً أســـوأ . إذ قـــد يمكنـــك تخصيـــص

سلسلة أحرف مكونة من ثمانى حروف وإستخدام الحروف الأربعة الأولى للتعبير عن السنة والحرفين التاليين للتعبير عن الشهر والحرفين التاليين للتعبير عن اليوم . بالإضافة إلى ذلك يمكنك كذلك إستخدام قيمة عدد صحيح لمتابعة عدد الأيام منذ عام 1900 أو منذ تاريخ إنشاء الشركة .

يوجد لدى كلا الطريقتين عيباً ألا وهو عدم وجود الدعم المتكامل للقيم من خلال وحدة الخدمة الخاصة بقاعدة البيانات . فإذا قمت بتخزين قيم التاريخ بمثابة سلسلة أحرف بدلاً من نوع بيانات خاص بالتاريخ - عندما تستخدم أداة خاصة بإستعلامات تفاعلية - فإن كل ما يظهر يتمثل فى قيمة مبدئية غير منسقة بالإضافة إلى ذلك ، عند إستخدام هذه القيم مع Visual Basic ، سوف تضطر إلى تحويل هذه القيم من وإلى متغير Date كى ما يتسنى لك الحصول على نطاق واسع من وظائف التاريخ والوقت الموجود بالفعل فى Visual Basic . وعلى المدى الطويل ، يعتبر إستخدام أنواع البيانات الخاصة بالتاريخ من خلال وحدة الخدمة الخاصة بقاعدة البيانات إجراء أفضل بكثير .

كيفية إختبار جمل SQL :

تتمثل إحدى مميزات لغة SQL فى إمكانية إستخدام اللغة نفسها بصورة تفاعلية كما يمكن كذلك وضعها داخل التطبيق . يعنى ذلك أنه يمكنك تكويد جمل SQL وإختبارها من خلال أداة إستعلامات تفاعلية ، ثم تقوم بعدئذ بإضافتها إلى البرنامج . رغم أن أدوات الإستعلامات تختلف من أداة إلى الأخرى وفقاً لنظام قاعدة البيانات الذى تستخدمة ، إلا أنها جميعاً تؤدى نفس الغرض . إذ أنك تقوم بإدخال الجملة وتنقر فوقها كى ما يتم تنفيذها . وسيتم عندئذ عرض النتائج على جهاز الكمبيوتر .

فيما يتعلق بالأمثلة المذكورة بالفصل الحالى ، سوف نستعين بقاعدة بياناتSQL Server أو Query Analyzer بغرض تشغيل الإستعلامات لإختبار قاعدة البيانات العينة الخاصة بالكتاب . ورغم ذلك ، ما أن تقوم بإنشاء وتحميل قاعدة البيانات العينة فى نظم قواعد البيانات المختار ، سوف تكون قادراً على إستخدام أداة الإستعلامات المتوافقة لتشغيل نفس الأمثلة .

كيفيـة إسـتخدام جملـة Select :

تعتبر جملة Select اكثر جمـل SQL المسـتخدمة شيوعـاً . ويتمثـل الغـرض مـن جملة Select فى تعريف الصفوف المطلوب إسـترجاعها مـن قـاعدة البيانـات .

وتعتبر جملة Select أكثر جمل SQL تعقيـداً . إذا يمثـل الـتركيب المذكـور أعـلاه جزءً صغيراً فحسب من التركيب الكامل لجملـة Select . غـير أنـه مـن النـادر أن تكـون بحاجة إلى أى شئ آخر بخلاف هذه الجمل عنـد إنشـاء التطبيـق . وتتكـون جملـة Select من سلسلة مـن الجمـل ، كعبـارة Form وعبـارة Where ومـا إلـى ذلـك إلا أن عبـارة From تعد العبارة الأساسية الوحيـدة لـدى جملـة Select .

وسنقوم الآن بدراسة كيفية عمل جملة Select الأساسـية ، ثـم نتنـاول عقـب ذلـك كلاً من الجمل التى تتعامل معـها .

جمل Select البسـيطة :

لإستخدام جملة Select ، كل مـا أنـت بحاجـة للقيـام بـه يتمثـل فـى تعريـف الجـدول والأعمدة المطلوب إسترجاعها من قاعدة البيانـات . إذ أنـه عقـب ذكـر الكلمـة الرئيسـية Select مباشرة ، يتم ذكر قائمة الأعمدة المطلوب إسـترجاعها ، ثـم تقـوم عبـارة From بعدئـذ بتحديـد إسم الجدول المطلوب الوصـول إليـه .

تعنى الجملة التالية بمهمة إسترجاع جميـع الأعمـدة مـن جـدول Customers الموجـود فى قاعدة البيانات العينة المتاحة على القـرص المرفـق بالكتـاب .

```
Select*
From    Customers
```

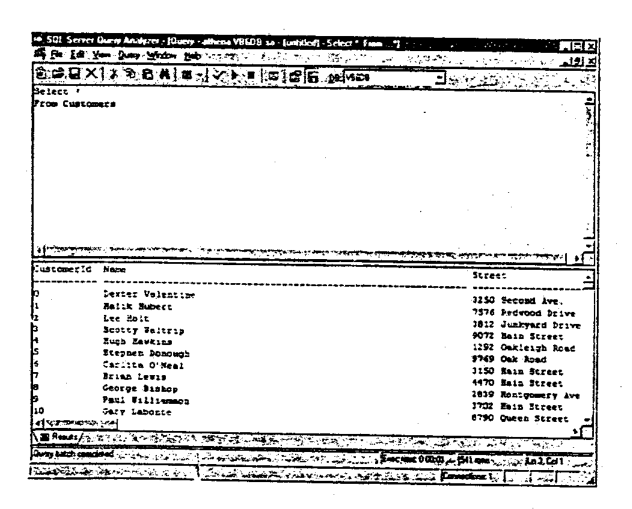
تعنى علامة (*) أنك ترغب فى إسترجاع كافـة الأعمـدة مـن الجـدول . مـن ثـم ينبغى أن ينشأ عن تشغيل هـذه الجملـة فـى أداة Query Analyzer نتـائج مماثلـة لتلـك الموضحة فى شـكل (١) .

كيفية إسترجاع قائمة من الأعمدة :

إذا كان كل ماتحتاج هو أعمدة معينة قليلة فحسب ، فينبغى عليك عندئذ إستبدال علامة (*) بقائمة من أسماء الأعمدة المطلوب ظهورها ، وذلك كما هو موضح على النحو التالى :

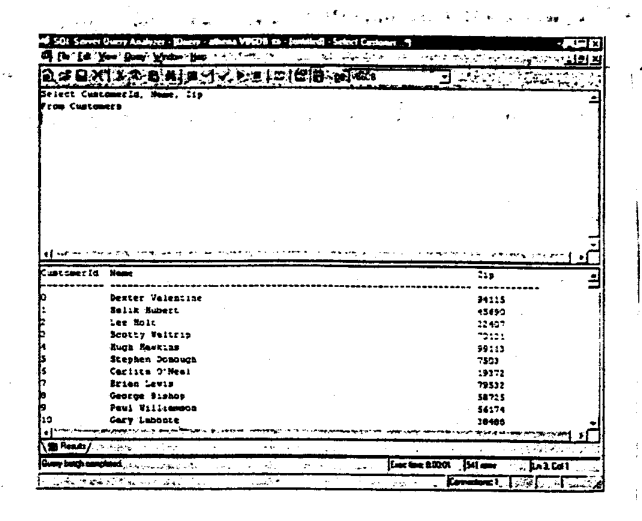
```
Select Customer , Name Zip
From Customers
```

بالإضافة إلى ذلك ، يجب فصل كل إسم عمود عن العمود السابق من خلاله فاصلة وذلك كما هو فى شكل (٢).



الشكل (١)

تشغيل إستعلام بسيط بغرض إسترجاع جميع الأعمدة من جدول Customer



الشكل (٢)

تشغيل إستعلام بسيط للحصول على قائمة من الأعمدة

كيفية تحديد مجموعة فرعية من الجداول :

إن ظهور الجدول بأكملة ليس أمرا مفيدا للغاية بالنسبة للبرنامج التطبيقي . فمن الطبيعي أنك سوف ترغب فى إسترجاع صف واحد أو عدة صفوف مرتبطة بقيمة أخرى فحسب . وهنا يأتى دور عبارة Where . إذ أن عبارة Where تسمح لك بإمكانية تحديد تعبير البحث الذى يقوم بتعيين مجموعة الصفوف المطلوب ظهورها .

ملحوظة : إذا كانت جملة Select تعتبر أكثر الجمل المستخدمة ذيوعا فى لغة SQL فإن عبارة Where تعد بمثابة أكثر العبارات المستخدمة شيوعا . إذ أنها تستخدم كذلك فى جمل أخرى بخلاف جملة Select ، بما فى ذلك جملتى Delete وUpdate

إستخدام تعبيرات البحث البسيطة :

تكمن الفكرة وراء إستخدام عبارة Where فى إنشاء تعبير بحث سوف يكون من شأنه أن ينتج الصف أو مجموعة الصفوف المطلوبة فحسب . على سبيل المثال ، دعنا نفترض إنك ترغب فى جميع المعلومات التى تدور حول العميل 431 . ويتم تخزين هذه المعلومات فى جدول

Customers . من ثم سوف يقوم تعبير البحث = 431 CustomerID بإسترجاع جميع هذه المعلومات . ونظراً لأن عمود Customerld يعتبر بمثابة Pri-mary Key بالنسبة إلى جدول Customers ، سوف يتم ظهور صف واحد فحسب من خلال جملة Select التالية ، وذلك كما هو موضح بشكل (٣) :

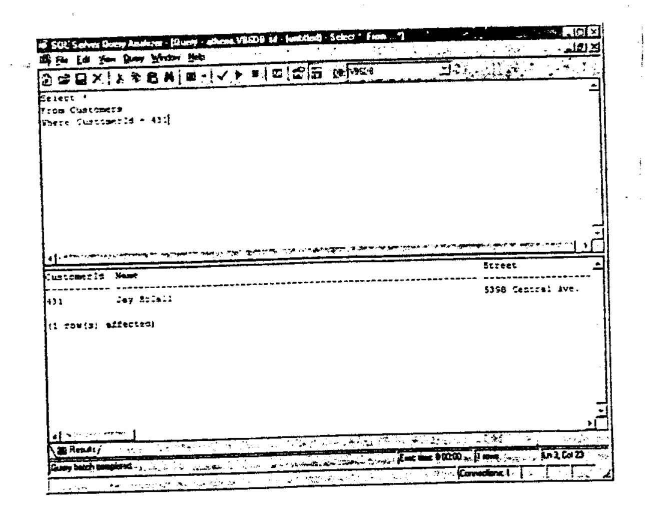
```
Select *
From Customers
Where Customer_Id = 431
```

ملحوظة : البحث عن التعبيرات التى توجد لديها المعاملات الخاصة : تدعم لغة SQL نفس المعاملات المستخدمة من قبل Visual Basic كافة ، بما فى ذلك (=و>و<و=>و=<و<> NOT و AND و OR) لتسهل من عملية إنشاء التعبيرات بالإضافة إلى ذلك ، تدعم لغة SQL كذلك القليل من المعاملات الأخرى مثل IN (سيتم مناقشتة فى جزء " الإستعلامات المتداخلة والفرعية " فيما بعد) وتعبير Like كى ما يتم التوافق مع نموذج معين ، والذى قد يحتوى على حروف بدل . ومما لاشك فيه أنه سيتم الإستعانة بالأقواس للتأكد أنه يتم وضع قيمة للتعبير كما ينبغى .

من ثم ، سوف تظهر الصفوف التى تحتوى على قيمة Customer_Id تساوى 431 ونظراً لأن Customer_Id يعتبر Primary Key الخاص بهذا الجدول ونظراً لأنك تدرك أيضاً أن كل قيمة من عمود Customerld قيمة مميزة ، لذا فلن يظهر إلا صفاً واحداً فحسب . لاحظ أنه إذا قمت بتحديد قيمة بالنسبة لعمود Customerld لاتوجد فى الجدول ، فلن تظهر أى صفوف .

مما لاشك فيه أنه إذا إستخدمت تعبيراً لصفوف متعددة ، يتم عندئذ ظهور صفوف متعددة . وقد ينشأ عن جملة Select التالية صفوف متعددة من جدول Custom-ers ، نظراً لأنه قد يكون هناك صفوف متعددة حيث يحتوى عمود State على القيمة " MD " ، وذلك على النحو الموضح فى شكل (٤) .

```
Select*
From Customers
Where State = " MD"
```



الشكل (٣)

كيفية تحديد معلومات عن CustomerId 431

```
Select *
From Customers
Where State = "MD"

CustomerId  Name                                  Street
----------  ----                                  ------
84          Fred Price                            3580 Maple Drive
205         Joseph Bell                           3428 Joppa Road
385         Kali Carlisle                         1609 Central Ave.

(3 row(s) affected)
```

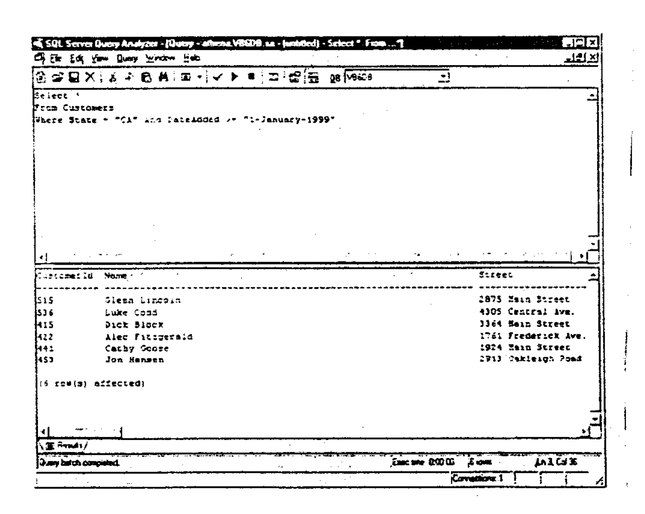
الشكل (٤)

كيفية تحديد العملاء القادمين من Maryland

إستخدام تعبيرات بحث أكثر تعقيداً :

يمكن لتعبيرات البحث أن تكون معقدة بقدر ما ترغب أن تكون . إذ يمكنك إستخدام AND و OR و NOT لإجراء عملية Compiling بالنسبة لبعض التعبيرات البسيطة المتعددة معاً بغرض تضييق نطاق البحث . وتؤدى جملة Select التالية إلى إظهار جميع العملاء الذين تم إضافتهم إلى قاعدة البيانات منذ عام 1999 والمقيمين فى كاليفورنيا كذلك ، وذلك على النحو الموضح فى شكل (٥).

```
Select *
From Customers
Where State = " CA" And  Data Added > = "1-Junuary – 1999 "
```



الشكل (٥)

إسترجاع جميع العملاء المقيمين فى كاليفورنيا الذين إضافتهم منذ عام 1999

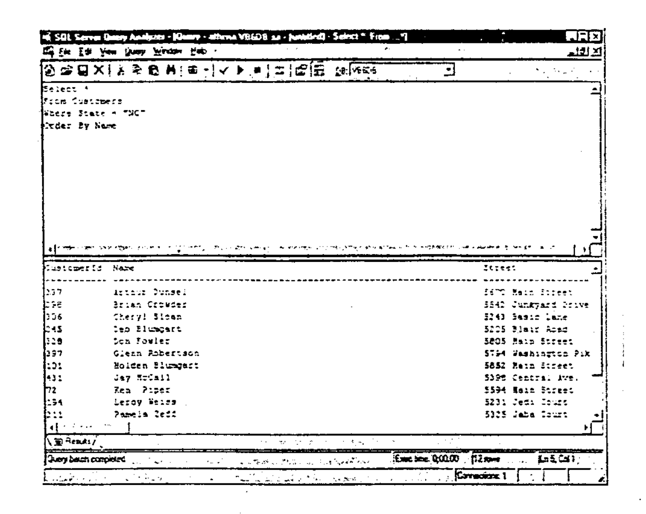
ملحوظة : حاول دائماً أن تضع عمود واحد على الأقل فى تعبير البحث بحيث يكون جزء من فهرس . وإلا ، سوف يتحتم على وحدة الخدمة الخاصة بقاعدة البيانات أن تقوم بعملية البحث من خلال كل صف موجود فى الجدول لإكتشافها الصفوف المطلوبة . رغم أن إجراء عملية البحث للجدول بأكملة قد يتم بصورة سريعة للحالية بالنسبة للجداول الضغيرة ، إلا أنه قد يستغرق وقتاً طويلاً بالنسبة للجدول الضخمة .

فرز النتائج :

كوضع إفتراضى ، لا ينتج عن جملة Select ظهور الصفوف بترتيب معين . وفى العديد من الحالات ، لايمثل ذلك الأمر ثمة مشكلة ، غير أنه إذا كنت ترغب فى عرض الصفوف للمستخدم ، فقد تجد أنه من المفيد أن يتم فرزها قبل عرضها من خلال عبارة OrderBy .

وفيما يتعلق بعبارة Order By ، فهى تلى عبارة Where وتحتوى على قائمة من الأعمدة المطلوب إستخدامها لفرز النتائج . فإذا كتبت تلو إسم العمود الكلمة الرئيسية Asc أو Desc ، فسيتم فرز ذلك العمود تصاعديا أو تنازليا على التوالى . أما إذا لم تقم بتحديد أى من الكلمتين الرئيسيتين ، فسيتم فرز البيانات على نحو تصاعدى . جملة المذكورة أدناه ، سيتم إسترجاع جميع العملاء المقيمين فى شمال كاليفورنيا وفرزهم عن طريق الإسم ، وذلك على النحو الموضح فى شكل (٦) .

```
Select *
From Customers
Where State = "NC"
Order By Name
```



الشكل (٦)

فرز الصفوف التى تم إسترجاعها من الجدول

لاحظ أنه سيتم فرز النتـــائج وفقـــا للإســـم الأول مـــن الشـــخص نظـــرا لأنـــه تـــم تخزيـــن البيانات فى حقل Name بحيث يتم تخزين الإسم الأول ثـــم الإســـم الأخـــير بعدئـــذ.

➢ كيفيـــة إســـتخدام الجـــداول المتعـــددة :

تسمح لك جملـــــة Select بإمكانيـــة الجمـــع بيـــن المعلومـــات مـــن خـــلال جـــداول متعددة داخل جدول " إفتراضى " واحد يتعذر تحيـــث الجـــدول " الإفـــتراضى " ذلـــك بيـــد أنـــه عندما تكون فى حاجة لتجميع معلومات مطلوب عرضـــها فـــى التطبيـــق فـــإن هـــذا الجـــدول يساعدك على إجراء هذه العملية بصـــورة أســـهل .

ملحوظة : يعرف المصطلح الفنى المتعلق بعملية الجمع بيـــن الصفـــوف والأعمـــدة فـــى جدوليـــن أو أكثر يعرف بإسم عملية الربـــط .

الإستعلامات المتداخلة :

تعتبر الإستعلامات المتداخلة من أكثر المهام التى يمكنك القيام بها فى جملة Select تعقيدا . فى الإستعلام المتداخل ، يمكنك إستخدام جملة select مرة ثانية (أو ثالثة أو أربعة) داخل الجملة الرئيسية . وتستخدم الجمل المتداخلة كى ما يتم إظهار مجموعة من القيم يمكن إستخدامها من خلال معامل I n .

ولتوضيح مفهوم الاستعلام المتداخل نسوق المثال التالى لبعض جمل الاستفسار المتداخل :

Book_No	Author	Book_Title	Pges	Publishing	Year	Price
7	Baron.s	Programming with Basic	386	Adison Weasly	1998	30
10	William. M	Data Base Concepts	440	Mcghill	1995	64
13	Roger Carter	Information Tevhnology	169	Oxford	1996	30
255	Amin .R	Digital Concepts	764	Adison Weasly	1990	365
549	Martin .W	The Information Edge	292	Adison Weasly	1986	139.35
550	Barry D	Data WareHouse	415	Mcghill	1997	229
553	Davies B.	Expert Database	150	Adison Weasly	1997	30
418	William .S	Operating System	701	N.J	1995	181.5
431	Hammer D.	Electronic Circuts	227	Adison Weasly	1991	93
422	Beaumont .A	Executic Logic Programs	995	Printchall	1995	98

Create a new tables which results from the following queries in SQL :

a)
```
Select Book_No,Book_Title,Price
From Book_Table
Where Price=(Select price
             From Book_table
             Where Book_No=13);
```

```
b)Select Book_No,Book_Title,Author,Price
  From Book_Table
   Where Pages>SOME
                    ( Select Pages
                       From Book_table
                       Where Year=1995);
C) Select Book_no,Book_title,Year
     From Book_table
     Where price =
               (Select  Price
                 From Book_table
                  Where Author ='Baron. S');
```

الحل :

تتمثل فكرة الاستعلام المتداخل فى أخذ نتيجة الاســـتعلام الأخـــير بجملـــة Select وجعلها بمثابة شرط للاستعلام الذى يسبقه سواء كان استعلام أساسى (أولى) أو إستعلام آخر فرعى . ويمكن توضيح ذلك على النحو التالى :

A)يلاحظ أن نتيجة جملة Select الأخيرة فى الاستعلام المتداخل للكتاب الذى رقمه " ١٣ " هو السعر " ٣٠ " والذى يمكن إعتباره بمثابة للشرط للاستعلام السابق اليه مباشرة وبذلك يكون جدول الاستعلام النهائى هو :

Book_No	Book_Title	Price
7	Programming with Basic	30
13	Information Tevhnology	30
553	Expert Database	30

B)يلاخظ أن نتيجة الاستعلام الأخير هو :

Pges
440
701
995

.. ويمكن أخذ تلك النتيجه كشرط للاستعلام الأصلى بتلك الجملة حيث يجب أن يزيد عـــدد صفحات الكتب عن البعض الموضح بالقائمة السابقة والناتجة من الاســتعلام الأخـير . وبذلــك يصبح جدول الاستعلام النهائى هو :

Book_No	Author	Book_Title	Year
255	Amin .R	Digital Concepts	1990

وهكذا يتم تطبيق نفـــس القــاعدة مــع الاســتعلام المتداخــل فــى المطلــوب (C) أو أى إستعلام متداخل آخر . مع مراعاة أنه قــد يكــون هنــاك أكــثر مــن إســتعلام فرعــى عــن الاستعلام الأصلـــى .

كيفيـة إسـتخدام الـدوال :

يمكنك وضع دوال مختلفـة فـى جملـة Select . ومـن أمثلـة تلـك الـدوال أو الجمل التعبـــيرات الرياضيـــة(...Count - Max - Min - Avg - Sum) والمثـــال التالى يوضح بكيفيـــة حســاب عــدد Customers المقيميــن فــى ولايـــة Maryland وذلك بإستخدام التعبير Count، وذلك على النحــو الموضــح فــى شــكل (٨) :

```
Select    Count(Customer_Id)
From    Customers
Where = State = " MD"
```

ويمكن أخذ تلك النتيجه كشرط للاستعلام الأصلى بتلك الجملة حيث يجب أن يزيد عـــدد صفحات الكتب عن البعض الموضح بالقائمة السابقة والناتجة من الاســـتعلام الأخـــير . وبذلـك يصبح جدول الاستعلام النهائى هو :

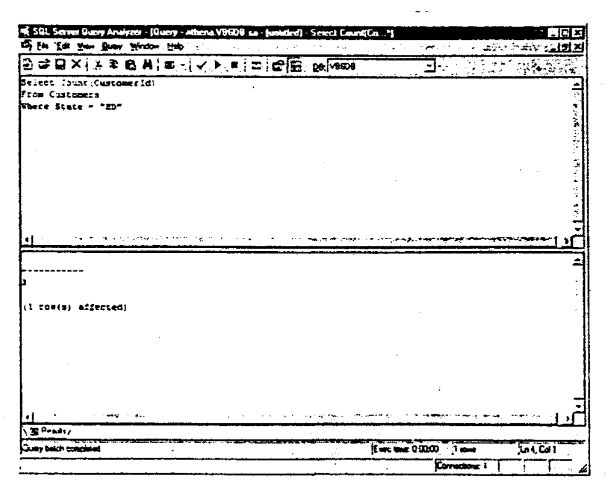
Book_No	Author	Book_Title	Year
255	Amin .R	Digital Concepts	1990

وهكذا يتم تطبيق نفـــس القـــاعدة مـــع الاســـتعلام المتداخـــل فـــى المطلـــوب (C) أو أى إستعلام متداخل آخر . مع مراعاة أنه قـــد يكـــون هنـــاك أكـــثر مـــن إســـتعلام فرعـــى عـــن الاستعلام الأصلـــى .

كيفية إستخدام الدوال :

يمكنك وضع دوال مختلفـــة فـــى جملـــة Select . ومـــن أمثلـــة تلـــك الـــدوال أو الجمل التعبـــيرات الرياضيـــة(...Count – Max – Min – Avg - Sum) والمثـــال التالى يوضح بكيفيـــــة حســـاب عـــدد Customers المقيميـــن فـــى ولايـــة Maryland وذلك بإستخدام التعبير Count، وذلك على النحـــو الموضـــح فـــى شـــكل (٨) :

```
Select    Count(Customer_Id)
From    Customers
Where = State = " MD"
```



شكل (٧)

حساب عدد العملاء المقيمين فى ولاية Maryland

وبالإضافـــة إلـــى ماســـبق ، يمكـــن أيضـــا إســـــتخدام الـــــدوال MIN و MAX و Average ، والتى من شأنها أن تقوم بحساب الحـــد الأدنـــى أو الحـــد الأقصـــى أو المتوســـط لقيم معينة وذلك عبر جميع الصفوف التى يتم تحديدها مـــن خـــلال قـــاعدة البيانـــات .

ويمكن عرض مثال آخر لتوضيح شكل تلك الدوال :

```
Select MIN(Salary) As min , Max(Salary)As max , AVG(Salary) As avg
From Staff_table
```

وفيما يلى نعرض جدول لتعريف تلك الدوال الخمس التالية :

COUNT	: Returns the number of values in a specified column
SUM	: Returns the sum of the values in a specified column
AVG	: Returns the Average of the values in a specified column
MIN	: Returns the smallest value in a specified column
MAX	: Returns the Largest value in a specified column

كيفية إدراج صفوف داخل الجـدول :

تستخدم جملة Insert فى لغة SQl لإضافـــة صـــف أو أكـــثر إلـــى الجـــدول .
ويتضح تركيب الجملة هذه كمـــا يلـــى :

```
Insert [Into] <table> [(Column>[,<column>}...] ) ]
[ Values (<value>[ , <value>]... ) ]
AS <Selectstatement> ]
Where
<table>is the name of where you want to insert new rows.
<Column> is the name of a column in the table.
<Value> is a value that you wish to insert into a column.
<Selectstatement> is a valid Select Statement .
```

تقوم جملة Insert بإضافـــة صـــف إلـــى الجـــدول المحـــدد . ويمكـــنك تحديـــد قائمة من الأعمدة التى سوف يتم تعيين القيم لـــها أو إســـتخدام قائمـــة مـــن الأعمـــدة التى تم تحديدها عندما تم إنشاء الجدول . إذ يمكنك تحديـــد قائمـــة القيـــم مـــن خـــلال

عبارة Value أو إستخدام الكلمة الرئيسية AS لتحديد جملة Select التى من شأنها أن تقوم باسترجاع القيم من جدول آخر .

من خلال عبارة value ، سوف تقوم بتحديد قائمة من القيم كى ما يتم إدراجها داخل الجدول ، بحيث يتوافق موقع كل قيمة مع ترتيب الأعمدة التى تم تحديدها فى جملة Insert ، أو فى حالة عدم تحديد قائمة الأعمدة ، فسوف تتوافق القيم مع ترتيب الأعمدة المذكورة فى تعريف الجدول .

على الجانب الآخر ، يسمح لك إستخدام عبارة AS مع جملة Select بإمكانية ملء الجدول بالبيانات من جدول آخر . كما هو الحال مع عبارة Value ، فينبغى أن تتوافق الأعمدة التى تم إسترجاعها من خلال جملة Select مع الأعمدة التى تم تحديدها بعد إسم الجدول .

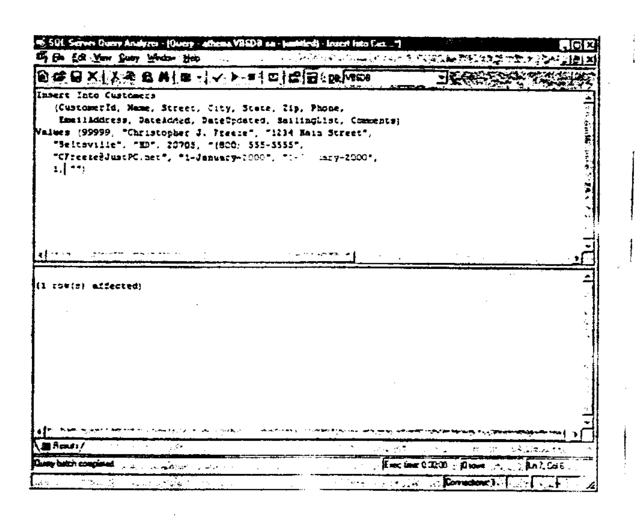
ملحوظة : عندما تجرى إختياراً للكود الذى من شأنه أن يقوم بحذف أو تحديث للبيانات فى أحد الجداول ، فإنه غالباً ما يكون إجراء مفيداً أن تقوم بإنشاء جدول مؤقت يحتوى على نسخة من بيانات الإختيار وإستخدام هذا الجدول لإجرء الإختبار . فإن ذلك يسمح لك بإمكانية تحديث البيانات بسهولة عقب قيام البرنامج بحذف المعلومات غير الصائبة إذ أن إستخدام جملة Insert مع عبارة AS يسهل كثيراً من القيام بالأمر .

جملة Insert البسيطة :

يتضح فيما يلى مثال لجملة Insert بسيطة للغاية :

```
Insert  Into  Customers
(Customerld , Name , street , City , State , Zip , Phone ,
EmailAdress , DateAdded , DateUpdated , Mailinglist, Comments)
Values (99999, ' Christopher J.Freeze', 1234 Main Street"
" Beltsville ", " MD ", 20705 , " (800) 555 – 5555 ",
" Cfreeze@JustPC.net ", ' 1–January–2000" , "1- January–2000",1"")
```

إذ تقــوم هــذه الجملــة بإضافــة صــف واحــد مــن الـ ' رمــات داخــل جـــــدول Customers وذلك كما هو موضح فــى شــكل (٨) . لاحــظ أننــى قمــت بتحديــد كــل مــن الأعمدة الموجودة بصورة تصريحية ، بحيث يتوافق كــل مــن القيــم المذكــورة فــى عبــارة Values مع العمود المذكور بعد إســـم الجــدو! .



الشكل (٨)

كيفية إضافة صف واحد داخل جدول Customers

تعتبر جملة Insert المذكورة أدنــــاه مماثلــة للجملــة الســابقة ، إلا أنــها تفــترض أن ترتيب الأعمدة الذى تم تعريفه فــى قــاعدة البيانــات هــو نفســه ترتيــب البيانــات فــى عبــارة Value .

```
Insert Into Customers
Values(99999, "Christopher J.Freeze ", "1234 Main Street ",
    " Beltsville " , " MD ", 20705 , " (800) 555-5555",
    " Cfreeze @ JustPC. Net ", " 1-January - 2000 ", 1-January-2000 " 1," ")
```

ملحوظة : التشغيل مرة واحدة أمر جيد ، أمـــا التشـــغيل مرتـــان فـــهو إجـــراء غـــير صـــائب يتسبب تشغيل هذه الجملة أكثر مـــن مـــرة واحـــدة فـــى حـــدوث خطـــأ . ويعـــزو السـبب فـــى ذلـــك إلـــى حقـــل CustomerId يمثـــل Primary Key الخـــاص بالجـــدول ، هذا فضلاً عن أن كل صـــف ينبغـــى أن يوجـــد لديـــه قيمـــة ممـــيزة واحدة فقط ، من ثم فإن محاولة إضافة صف آخـــر يحتـــوى علـــى نفـــس القيمـــة تسفر عن وقوع خطـــأ .

كيفية حـذف الصفـوف مـن الجـدول :

يتم إستخدام جملة Delete لإزالة صف أو أكـــثر مـــن الجـــدول . ويتضـــح فيمـــا يلى تركيب جملـــة Delete :

```
Delete From <table>
[Where <expression>]
```

Where

<table> is the name of the database table from which you want to delete the rows .

<expression> is an expression that is used to determine which rows to delete .

تعتبر جملـــة Delete الجملـــة المقابلـــة لجملـــة Insert . إذ يتـــم إســـتخدامها بغرض حذف صفـــوف مـــن الجـــدول . تســـتخدم جملـــة Delete عبـــارة Where مـــن جملة Select كى ما تعريف الصفوف التـــى ســـيتم حذفـــها .

تنبيه : حذف الصفـــوف : عنـــد حـــذف صـــف معيـــن مـــن جـــدول ، يتعيـــن عليـــك إســـتخدام Primary Key فى عبارة Where لتعريف الصـــف المحـــدد المطلـــوب حذفـــه.

مثال على جملة Delete:

سوف تقوم جملة Delete التالية بحذف الصف الذى تم إضافته منذ قليل ، وذلك كما هو موضح فى شكل (٤-١٠) .

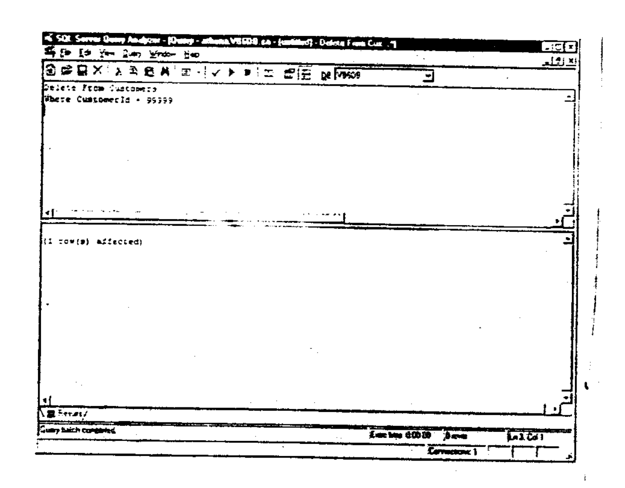
```
Delete From Customers
Where CustomerId = 99999
```

يتم تحرير الكود الخاص بعبارة Where بنفس الأسلوب التى تم من خلاله تحرير ، كود جملة Select . إلا إننى فى هذه الحالة أرغب فى حذف صف واحد فحسب ، لذا فإننى بحاجة إلى تحرير الكود الخاص بعبارة Where لتحديد الصف المطلوب حذفه.

من ناحية أخرى ، قد تمثل جملة Delete أيضاً مشكلة كبيرة للغاية . فالجملة التالية سوف تقوم بحذف كافة الصفوف الموجودة فى جدول Customers :

```
Delete From Customers
```

لاحظ أن الفارق الوحيد بين هذه الجملة والجملة السابقة يتمثل فى عدم وجود عبارة Where .



الشكل (٩)

حذف صف واحد من جدول Customers

كيفية تحديث الصفوف فى الجدول .

تسمح لك جملة Update بإمكانية تغيير القيم فى عمود أو أكثر داخل صف أو أكثر . ويتضح فيما يلى تركيب جملة Update :

Update <table>

Set<column> = <value > [,<column>=<value>]...

Where <expression>

Where

<table> is the name of the table you want to update .

<column> is a Column name in the table you want to update.

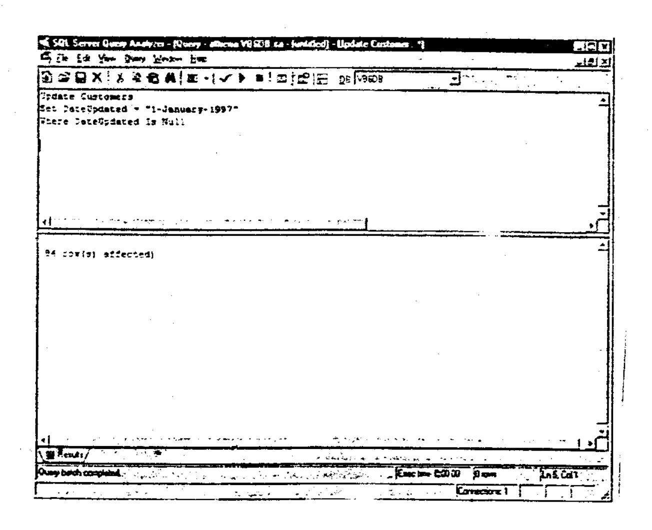
<value>is an expression containing the new value for the column.

<expression>is true for the rows you want to update in the table .

تتيح لك جملة Update إمكانية تغيير أية قيمة داخل أى صف موجود بالجدول . وكما هو الحال مع جملة Delete نكون بحاجة إلى ضم عبارة Where لقصر تأثيرات هذه الجملة على الصفوف المطلوب تحديثها فحسب وإلا ، فسينطبق هذا التغيير على كافة الصفوف الموجودة فى الجدول .

مثال على جملة Update:

سوف تقوم جملة Update التالية بالبحث عن جميع الصفوف التى يوجد لديها قيمة Null بالنسبة إلى DateUpdated فى جدول Customers وإستبدالها بتاريخ صالح للإستخدام ، وذلك على النحو الموضح فى شكل (١٠) .



الشكل (١٠)

تغيير قيم Null إلى تاريخ صالح للإستخدام

➢ كيفية إستخدام جملة Create Table :

تستخدم جملة Create Table لإنشاء جدول جديد فى قاعدة البيانات .

ويتضح فيما يلى التركيب الخاص بعبارة Create Table :

Create Table <tablename>(<columndef>[,<columndef>]...) where
<columndef>:: = <columnname><datatype>[Null][NotNull]
and
<tablename> is the name of your table.
<columnname> is the name of a column in your table.
<datatype> is avalid data type .

تسمح لك جملة Create Table بإمكانية تعريف مجموعة من الأعمدة التى تؤلف جدولا . غير أنه لابد ألا يوجد الجدول بالفعل فى قاعدة البيانات ، أو أنك ستحصل على رسالة تنبيه عن وجود خطأ حينما تحاول أن تقوم بإنشائه .

وسوف تقوم الجملة التالية فـــى لغـــة SQL بإنشـــاء جـــدول Customers لقـــاعدة بيانات SQL Server:

```
Create Table Customers(Customerld Int Not Null,
Name Varchar (64) , Street Varchar (64) , City Varchar(64),
State Char(2) , Zip Int , Phone Varchar(32)
EmailAddress Varchar (128) , DateAdded Datetime,Mailinglist Bit
Comments Varchar (256)
```

لاحظ أننا قمنا بتعريـــف قيمـــة Customer_Id بمثابـــة Not Null نظـــرا لأن هـــذا العمود يمثـــل Primary Key للجـــدول .

كيفيـة إســتخدام جملـة Create Index :

تستخدم جملة Create Index لإضافة فهرس إلى جدول فـــى قـــاعدة البيانـــات

ويتضح فيما يلى تركيـــب جملـــة Create Index:

```
Create [unique] Index < undexname > On <tablename>
(<columnname>['<columnname>]...)
Where
<indexname>is the name of your index.
<tablename > is the name of your table.
<columnname> is the name of a column in your table.
```

تقوم جملـــة Create Index بإضافـــة فـــهرس إلـــى جـــدول باســـتخدام الأعمـــدة المحددة . فيمكن أن يـــؤدى إســـتخدام الفـــهرس إلـــى تحســـين أداء الإســـتعلامات التـــى تستخدم هذه الأعمدة على حساب المزيد من العمـــل الـــذى ينبغـــى أن تقـــوم بـــه قـــاعدة البيانات فى كل مرة يتم فيها تغيير أية قيمة فـــى الأعمـــدة المحـــددة .

من ناحيـــة أخـــرى ، يعنـــى إســـتخدام الكلمـــة الرئيســـية Unique أن مجموعـــة القيم الموجودة فى الفهرس سوف تصبح مميزة فى هـــذا الجـــدول . وبعـــد ذلـــك الآمـــر سمة مفيدة إذا كنت بحاجة أن تتـــأكد أن لا يوجـــد لديـــك صفيـــن أو أكـــثر بـــهما نفـــس القيمة فى الأعمدة المفهرســـة .

ملحوظة : لإزالة فهرس من الجدول يتعين عليك إســـتخدام جملـــة DropIndex <إســـم الفهرس> فى لغة SQl .

مثال على جملة Create Index :

تقــوم جملـــــة SQL التاليـــــة بإنشـــــاء فـــــهرس Unique علـــى حقـــل Customer_Id الخـــاص بجـــدول Customers :

Create Unique Index Customer Index
On Customers(Customer_ld)

من ثم ، يضمن ذلك الفـــهرس أن كـــل قيمـــة موجـــود فـــى حقـــل Customerld فى الجدول سوف تكون قيمة مميزة فضلاً عـــن تشـــغيل الإســـتعلامات التـــى تســـتخدم عمود Customer_ld فى عبـــارة Where بصـــورة أســـرع .

كيفيـــة إســتخدام جملـة Create View :

تقوم جملة Create View بإنشـــاء جـــدول إفـــتراضى يمكـــن إســـتخدامه مثـــل أى جـــدول آخر فى قاعدة البيانات . ويتضح فيما يلى الـــتركيب الخـــاص بـــهذه الجملـــة :

Create View <viewname> [(columnname>[,<columnname>]..)]
As <selectstatement>
Where
<viewname> is the name of your view.
<columnname> is the name of a column in your view.
<selectstatment> is valid select statement that returns the information in your view.

ولايمكن التمييز بيـــن الجـــدول الإفـــتراضى الـــذى تقـــوم جملـــة Create View بإنشائه فى قاعدة البيانات عن الجـــدول العـــادى فيمـــا يتعلـــق بأيـــة عمليـــات تتضمـــن جملة Select . إذا أنه غالباً مـــا يمكـــن أن يتـــم تحديـــث الجـــدول الإفـــتراضى ، وذلـــك وفقاً إلى كيفية إنشاء طريقـــة العـــرض .

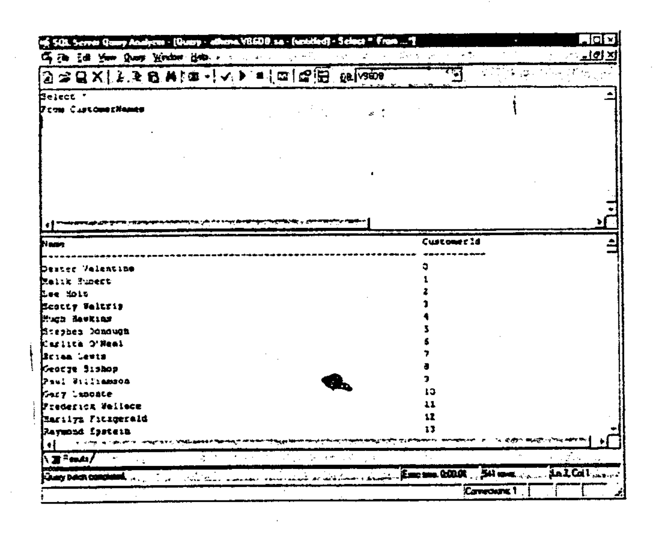
لتحديث طريقة العرض ، ينبغى أن تشير جملة Select إلى جدول واحد فقط يطلق عليه الجدول الأساسى ، ولاينبغى أن ينشأ عنها أية قيم حسابية بإستخدام الدوال و/أو الصيغ الرياضية . بالإضافة إلى ذلك ، ينبغى أن تسمح أى أعمدة غير موجودة بصورة تصريحية فى طريقة العرض بوجود قيم Null . من ثم ، عندما تحاول إضافة صف إلى طريقة عرض ، سيتم ضبط أى أعمدة فى الجدول الأساسى لاتعتبر جزء من طريقة العرض على Null .

ملحوظة : لإزالة طريقة العرض من الجدول ، يتعين إستخدام جملة Drop View .إسم طريقة العرض, فى لغة SQL .

وتقوم جملة SQL التالية بإنشاء طريقة عرض تتكون من إسم العميل وعمود Customer_Id :

```
Create view Customer_Name As
Select Name , Customer_Id
From Customers
```

ويوضح شكل (١١) نتائج إستخدام جملة Select إزاء طريقة العرض . ويمكن تحديث طريقة العرض ، نظراً لأن الأعمدة التى لم يتم ضمها سوف تسمح بوجود قيم Null . ويطلق على هذه التقنية مصطلح التقسيم الرأسى Vertical Partitioning ، نظراً لأن بعض الأعمدة فحسب هى التى ستكون متاحة للمستخدم .



الشكل (١١)

منع بعض الأشخاص من مشاهدة بعض المعلومات الخاصة بالعملاء

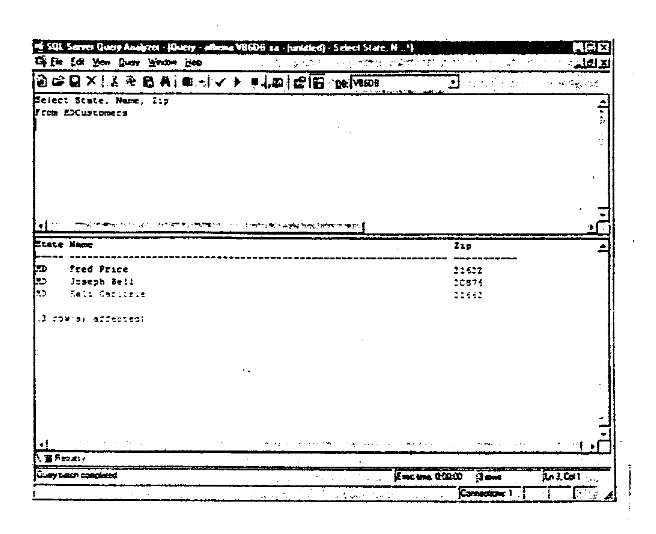
بخلاف إسم العميل و Customer_Id

من ناحية أخرى ، يمكنك كذلك إستخدام عبارة Where ليتم إسترجاع بعض الصفوف الموجودة فى الجدول فحسب . ويطلق على هذه التقنية التقسيم الأفقى Horizontal Partitioning . ويعد التقسيم الأفقى مفيداً إذا كنت بحاجة لإنشاء طريقة عرض حيث يتم إسترجاع بعض الصفوف من الجدول فحسب . وتقوم جملة SQL التالية بإنشاء طريقة عرض تحتوى على أسماء العملاء المقيمين فى Maryland فقط ، وذلك على النحو الموضح فى شكل (١٢) .

```
Create view MdCustomers As
Select *
From Customers
Where State = "MD"
```

لاحظ أن طريقة العرض هنا قابلة للتحديث ، نظراً لوجود جميع الأعمدة الخاصة بالجدول الأساسى . فضلاً عن ذلك ، رغم أن طريقة العرض محدودة كى ما يظهر أسماء العملاء القادمين من Maryland فحسب ، إلا أنه يمكنك إدراج صفوف

باستخدام قيمة مختلفة بالنسبة إلى State . من ثم ، يتسنى لك أن تقوم بإضافة صف إلى طريقة العرض والذى لن تتمكن من إسترجاعه لاحقاً .



الشكل (١٢)

إقتصار طريقة العرض على أسماء العملاء القادمين من Maryland فقط

الخلاصة :

يطلق على جمل Create Table و Create View و Create Index جمل Data Defination Language (DML) ، بينما تعرف جمل Inser و Delete و Vpdate بمثابة جمل Data Mamipulation Language(DML) فى معظم نظم قواعد البيانات المتاحة فى هذه الآونة ، نادراً ما تقوم بتنفيذ جمل DDl حينما ترغب فى إنشاء بنية قاعدة بيانات . إنما تستخدم بدلاً من ذلك أداة متاحة مع نظام قاعدة البيانات تسمح لك بإمكانية ملء جميع المعلومات فى الجدول أو تستعين بمعالج يساعدك فى إنشاء الجدول أو الفهرس .

لايعنى ذلك أنه لايتم إستخدام جمل DDL . وإنما يعنى أنك تقوم بإدخال المعلومات بأسلوب مختلف فحسب . إذ أن الأداة الخاصة بقاعدة البيانات تضم سمة عادة ما تسمح لك بإمكانية إنشاء جمل SQL من خلال التعريفات التى قمت بإدخالها . ثم قد تستخدم جمل SQL هذه بعدئذ لإنشاء نسخة من قاعدة البيانات على جهاز كمبيوتر آخر أو لوضعها فى التطبيق إذا ترغب أن يكون المستخدمين قادرين على إنشاء بينات قواعد بيانات بصورة سريعة . من ناحية أخرى ، تضم لغة Visual Basic العديد من أنماط العناصر التى من شأنها أن تجعل المبرمج لا يستعين بجمل SQL . من ثم رغم أنك قد لاتكون بحاجة لمعرفة كيفية إستخدام عبارات SQL لكتابة تطبيق قاعدة بيانات ، إلا أنك قد تجد نفسك قد تاجوزت عن معلومات لها كبير الأهمية فى ظل وجود هذه التكنولوجيا .

من ناحية أخرى ، سوف يساعدك إستخدام أداة الإستعلامات التفاعلية بغرض إسترجاع المعلومات من قاعدة البيانات على معرفة ما إذا كان البرنامج يعمل كما ينبغى أم لا . إذ يمكنك إستخدام جملة Select لإسترجاع صفوف من جدول ، هذا فضلاً عن أنه يمكنك التأكد ما إذا تم تحديثها بالصورة الصحيحة أم لا . وعند إستخدامها مع دالة Count ، يمكنك إكتشاف ما إذا كان البرنامج قد قام بمعالجة العدد الصحيح من الصفوف أم لا .

بالإضافة إلى ذلك ، يمكن إستخدام جملة Insert وجملة Create Table معاً لإنشاء نسخ إختبار من الجدول ، مما يساعدك على إجراء إختبار لصفوف

التحـــديث والحذف بصورة متكررة حتـــى تتـــأكد أن البرنـــامج يتـــم تشـــغيله كمـــا ينبـــغ . تقدم معظم الشركات المنتجة لقواعد البيانات بيئـــة متطـــوره لتنفيـــذ الجمـــل الخاصـــة بلغـــة SQL . من خلال هذه البيئة ، من الممكن أن يتـــم تحريـــر برامـــج تحتـــوى علـــى عبـــارات متعددة من لغة SQL ويطلق عليها الإجـــراءات المخزنـــة ، والتـــى مـــن شـــأنها أن تـــؤدى العمليات المعقدة للغاية . ونظراً لأنه يتـــم تشـــغيل الإجـــراءات المخزنـــة كليـــة علـــى وحـــدة الخدمة الخاصة بقاعدة البيانات فقد يتم تشغيلها بصورة اســـرع بشـــكل كبـــير عنـــها مـــا إذا تم تنفيذها جملة واحدة كل مرة من خلال الكمبيوتر المحلـــى . ويعتـــبر هـــذا المفـــهوم مـــهماً حيث ينبغى وضعه فى الإعتبار ، لاسـيما حينمـــا تقـــوم بـــإجراء عمليـــة تطويـــر للتطبيقـــات الخاصة بقاعدة البيانات ، وسوف نتطرق إليه بمزيـــد مـــن التفـــاصيل لاحقـــاً عندمـــا نتنـــاول موضوع الإجراءات المخزنـــة .

علاوة على ذلك ، ضع فى إعتبـــارك أن العديـــد مـــن مـــزودى OLE DB يقومـــون بترجمة الأنشطة التى تقوم بها إلى الجمل الخاصـــة بلغـــة SQL ، ثـــم يتـــم إرســـالها بعدئـــذ إلى وحدة الخدمة الخاصة بقاعدة البيانات كى يتم تنفيذها . ويحـــدث ذلـــك حتـــى فـــى حالـــة إذا لم تقم بوضع الجمل الخاصة بلغـــة SQL داخـــل البرنـــامج بصـــورة تصريحيـــة . ومـــن ثم ، فإنك قد تفضل تكويد الجمـــل الخاصـــة بلغـــة SQL بنفســـك بـــدلاً مـــن إســـناد هـــذه المهمة إلى المزود فى بعض الحالات التى تتســـم بالدقـــة المتناهيـــة .

هناك شئ آخر ينبغـــى وضعـــه فـــى الإعتبـــار ، ألا وهـــو أن الفصـــل الحـــالى يقـــدم معلومات كافية لكى تبدأ فى تعلم كيفيـــة إســـتخدام لغـــة SQL . إلا إننـــى أعتقـــد أن أفضـــل طريقة للتعلم تتحقق من خلال العمارسة العملية ، مـــن ثـــم فـــإننى أقـــترح عليـــك أن تتـــدرب مـــدة كافيـــة علـــى كيفيـــة إســـتخدام Query Analyzer أو SQL *Plus أو أى برنـــامج SQL تفاعلى آخر لتطبيـــق كيفيـــة إنشـــاء وتنفيـــذ الجمـــل الخاصـــة بلغـــة SQL بصـــورة علمية . ورغم أننى أوثر إستخدام الجمـــل البســـيطة بصـــورة دائمـــة ، إلا أن هنـــاك بعـــض الحالات التى يفضل فيها إستخدام العبارات المعقدة للغايـــة فـــى لغـــة SQL . ومـــن الجديـــد بالذكر أن أفضل وسيلة لتحرير الجمل المعقدة للغاية فـــى لغـــة SQL بحيـــث تعمـــل بصـــورة صحيحة تتمثل فى التدريب على كيفية تحير الجمـــل البســـيطة فـــى لغـــة SQL .

فى النهاية نستخلص أنه من خلال هذا الفصل قد تعلمنا ' ـاط التالية :

- الجمل الخاصة بلغة SQL وانــواع البيانــات .
- كيفية إستخدام جملة Select لإسترجاع المعلومــات مــن قــاعدة البيانــات .
- كيفيــة إســتخدام جمــل Insert و Delete و Update لمعالجــة البيانـــات الموجودة فى قاعدة البيانـــات .
- كيفية إستخدام جملة Create Table لإنشــاء جــدول جديــد .
- كيفية إستخدام جملة Create Index كــى تســمح لقــاعدة البيانــات بــالوصول إلى صفوف بعينها فى الجدول على نحو أكــثر ســرعة.
- كيفيــة إســتخدام جملــة Create View لإنشــاء جــدول إفــتراضى يمكــــن إستخدامه تمامــاً مثــل الجــدول الحقيقــى ، بيــد أن محتوياتــه يتــم إنشــائها بصورة ديناميكية من خلال جداول أخرى فــى قــاعدة البيانــات .

الفصل السادس

قواعد البيانات الموزعة وشبكات الاتصال

الفصل السادس

قواعد البيانات الموزعة وشبكات الاتصال

أتخذت العديد من المشـــروعات والمنشـــأة فـــى الفـــترة الأخـــيرة سياســـة تركيـــب عدة حاسبات فى أماكن متفرقة منها وربطها بوســـائل إتصـــال . وقـــد يتـــم ذلـــك بحواســـب شخصية موصلة محلياً ببعضها البعض داخل موقـــع واحـــد أو حتـــى فـــى نفـــس المكتـــب أو يتم ذلك بمجموعة حاسبات صغيرة وكبـــيرة موزعـــة علـــى نطـــاق جغرافـــى واســـع تربـــط بينها بوسائل الإتصال . وسوف نعطى مـــن خـــلال هـــذا الفصـــل مفـــهوم النظـــم الموزعـــة بشكل عام ونموذج توزيع الطاقـــه الكمبيوتـــر داخـــل المنشـــأة والربـــط بينـــها مـــن خـــلال شبكات الإتصال . ودور قواعد البيانات الموزعـــة فـــى ربـــط هـــذه الشـــبكات ببعضـــها مـــن أجل توفير المعلومات اللآزمة للمســـتخدمين المتعدديـــن والمختلفيـــن .

(١) فكرة النظم الموزعة:

يطلق مصطلح النظام الموزع علـــى مجموعـــة متنوعـــة مـــن أنظمـــة الحاســـبات المادية التى تقع فى أماكن متباعدة جغرافيا وترتبط معـــا بوســـائل إتصـــال إلكترونيـــة.

وهناك العديد من الإعتبارات يجب أخذها فى الحسبان بالنسبة للنظم الموزعة :

- أماكن المعالجة وأنواع العلاقة المتبادلـــة بينـــها .
- مكان تخزين البيانات وكيفية تقديـــم البيانـــات للمســـتخدمين .
- طبيعية الربط بين الأمـــاكن المختلفـــة .
- المعايير القياسية المطبقة فـــى الإتصـــالات .

ونقدم فيما يلى شرحاً موجزاً لهذه النقاط:

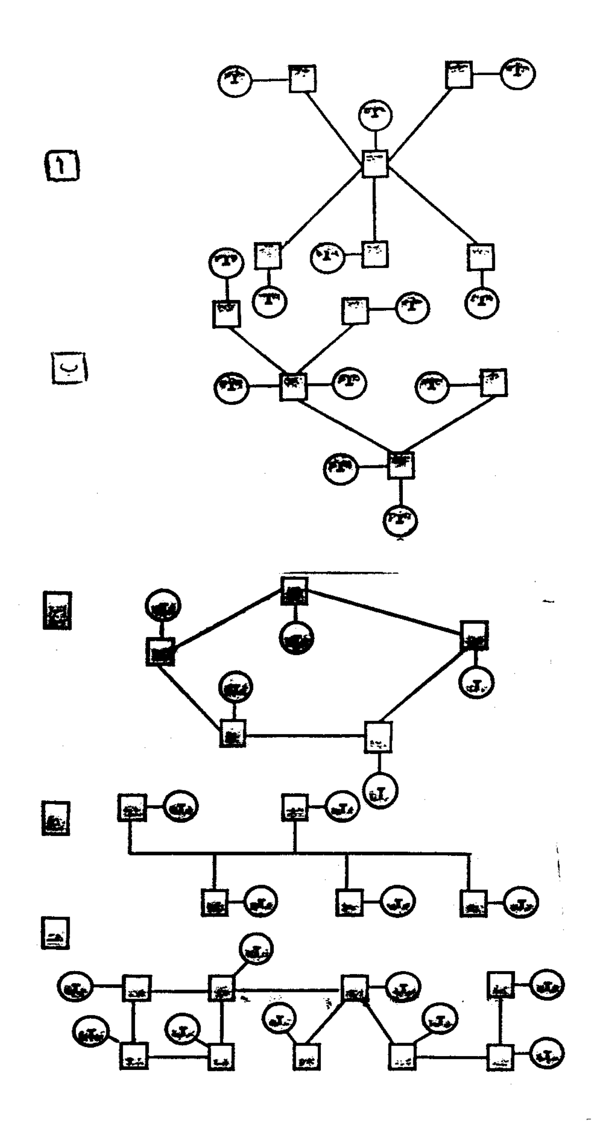
١- تتحقق المعالجة الموزعة Distributed Processing بعده طرق ، ابسطها أن تقوم الأجهزة باجراء المعالجة الخاصة بها ، وتتبادل رسائل فيما بينها من خلال نظام بريد . وتتحقق درجة أعلى من التكامل بما يسمى " المعالجة التعاونية Cooperative Processing " ، حيث يقوم معالجان متباعدان جغرافيا بعملية المعاجلة سويا ، فيقوم واحد منهما بارسال مخرجات معالجته للآخر لكى يقوم باكمال المعالجة . ويكون الوضع فى هذه الحالة أكثر تعقيداً إذا كانت نظم التشغيل مختلفة ، وهنا يجب الإستعانة بنظم تشغيل تعاونية Cooperative operating systems .

٢- وتوجد قواعد البيانات الموزعة distributed databases فى حالة حفظ البيانات فى أكثر من موقع ، ويمكن أن يحدث ذلك بطريقتين . فيمكن الإحتفاظ بملف البيانات " الرئيسى " فى موقع معين ، ويتم إرسال نسخ منه إلى كافة الموقع ، وفى هذا النظام غير مسموح لأحد المواقع بتغيير بيانات موقع آخر محلياً ، وإلا أصبحت البيانات فى جميع المواقع غير متطابقة . وفى حالة تغيير البيانات ، يتم التحديث فى الملف الرئيسى ، والذى ترسل نسخ منه للمواقع الأخرى صباح اليوم التالى .

إلا أن قواعد البيانات الموزعة بالمعنى الحقيقى فتكون على صورة توزيع البيانات على عدة مواقع دون نسخ لها . ويمكن لكل موقع أن يحصل على ما يشاء من بيانات فى أى موقع آخر عن طريق برامج قواعد البيانات والإتصالات . فمن وجهة نظر المستخدم تكون كافة البيانات متاحة له . وفى كلتا الحالتين ، تكون عملية التنميط لتعريف البيانات أمراً جوهرياً .

٣- أما طبيعة خطوط الاتصال communication links فتختص بطرق التوصيل بين المواقع (طوبوغرافية الشبكة toplogy) ، وتقنية تصميمها . ويبين شكل رقم (١) أنواعا مختلفة من طبوغرافيات الشبكات تخدم مختلف الأغراض. فالطبوغرافية الهرمية Hierarchical مثلا تخدم المنظمات التى حاسب مركزى كبير الحجم فى المقر الرئيسى ، موصل بشبكة عالمية بعدة حواسب صغيرة فى مواقع محلية ، متصلة بدورها بحاسبات شخصية محتاجة الى أن تتصل ببعضها البعض . أما الطبوغرافية الحلقية Ring أو الخطية Bus فشائعتان فى الشبكات المحلية للحاسبات الشخصية . وهناك أخيرا الطبوغرافية المختلطة Hybrid والتى يتم فيها ربط عدة أنظمة ببعضا البعض . وبالاضافة لطبوغرافية الشبكة فان التقنيات المستخدمة فى مكونات الشبكة ، كالموصلات وكروت الرقائق الدقيقة والبرمجيات المستخدمة فى إدارة الاتصالات فى الشبكة ، فجميعها لها دور فى تحديد خصائص الشبكات .

٤- وأخيرا لدينا المعايير القياسية Standards التى تحكم الطريقة التى تتخاطب بها الأجهزة فيما بينها مثل معايير Open system Interconnection (OSI) ومعناها " إتصالات النظم المفتوحة" .



شكل رقم (١) يوضح طبوغرافيات مختلفة للشبكات (أ) هرمية ب)نجمية ج) حلقيـــة د) خطيــة هـ) مختلطـة)

٢) فوائد النظم الموزعة:

كان ظهور النظم الموزعة لأول مرة فى السبعينيات ، ثم شاع إستخدامها بعد ذلك ، وذلك بسبب التطورات التقنية فى وسائل الإتصالات وقواعد البيانات الموزعة وبرمجيات الإتصالات وأيضاً بسبب إكتشاف الفوائد التى تعود على المنشآت والمنظمات من إستخدام هذه النظم وتتمثل أهم الفوائد التى تعود على المنشآت من النظم الموزعة فى :

إرضاء أكثر للمستخدم : فى النظم المركزية يشعر المستخدمون بأنهم منعزلون عن مركز الحاسب وفريق العاملين به من خبراء ومتخصصين خاصة إذا كانوا متباعدين جغرافياً . ويؤدى هذا الإنعزال إلى ضعف الإستجابة لمتطلبات المستخدمين الحقيقة ، وبطء وصول المساعدة حين الحاجة إليها من خلال الطرفيات . ولاتوجد هذه المشاكل فى نظم المعلومات المحلية ، لأن المستخدمون هنا يشعرون بإستقلالية أكبر ومن الضرورى طبعاً ألا تكون هذه الإستقلاليه مطلقة ، بل يجب دوام الإتصال بالمركز ، ليس فقط لتقاسم البيانات ، بل لضمان سير العمل فى كافة المواقع بإنسجام لتحقيق الأهداف المشتركة ، غير أن ذلك لا يلغى سلامة مبدأ الإستقلالية ، فالنظم الموزعة تضمن أن يتم الإتصال بالمركز ، مع ضمان الإستقلالية وإرضاء المستخدمون فى نفس الوقت.

مرونة وضع وتطوير النظم : تستطيع المنشآت النامية أن تتوسع بإستمرار كلما دعت الحاجة عن طريق التوسع فى نظامها الموزع . أما فى النظام المركزى فالنمو يعنى زيادة التحميل على النظام الحالى ، وقد يصل الأمر إلى الوصول بالنظام إلى أقصى طاقة له ، مما يوجب إما تغييره بالكامل وهو أمر قد يكون مكلفاً ومسبباً للخسارة ، أو صرف النظر عن التوسع .

تكلفة أقل للإتصالات : تتم فى النظم الموزعه كافة العمليات الحاسوبية المحلية فى الموقع ، ولايتم الدخول فى شبكة إلا عندما تكون البيانات مطلوبة لمكان آخر ، وبذلك تقل تكلفة الإتصال بالمقارنة بالنظم المركزية التى تتطلب نقل كافة البيانات من كل المواقع ليتم معالجتها مركزياً .

تلافى التعطل الشامل : إن حدوث عطل فى نظام مركزى قد يعنى تعطل النظام ككل ، مع ما يسببه ذلك من مشاكل جمة ، ولكن فى النظام الموزع لايؤثر عطل فى موقع

على بقية المواقع فتستمر فى عملها ، بل وقد يمكنها أن تحـل محـل الموقـع المعطـل فـى بعض عملياتـه .

**** تحسن زمن الإستجابة** : تعانى النظم المركزيـة مـن إزدحـام البيانـات وقـت الـذروة ، مما يجعل إستجابة الحاسوب المركـزى للمسـتخدمين بطيئـاً .

ومع الإقتناع هذه الفوائد ، فإن هنـاك مشـاكل مرتبطـة بتوزيـع الأنظمـة يجـب عدم إغفالها عند تقييم الإستراتيجية العامـة للنظـم :

١- **عدم وجود تنظيم معيارى ورقابة مركزية** : مـن الطبيعـى أن ينفـرد المسـتخدمون المحليون بطرقهم الخاصة التى تتناسب مع ظروف علمـهم ، وقـد يـؤدى ذلـك إلـى صعوبـة الإتصال والتعاون بين أجـزاء المنظمـة .

٢- **الحاجة لبرمجيات معقدة** : وهى البرمجيات التى تقـوم بـإدارة إتصـالات الشـبكة ، أو نظـم قواعد البيانات الموزعـة .

٣- **إحتمال تكرر البيانات** : إن تعدد البيانات المشـتركة بيـن المواقـع قـد يكـون لـه فوائـد متمثلة فى تقليل تكلفة الإتصالات وزيـادة سـرعة الإسـتجابة ، ولكـن قـد يـؤدى فـى نفـس الوقت إلى عدم توافق البيانات بين المواقع إذا تم تحديثـها فـى أحدهـا دون الآخريـن .

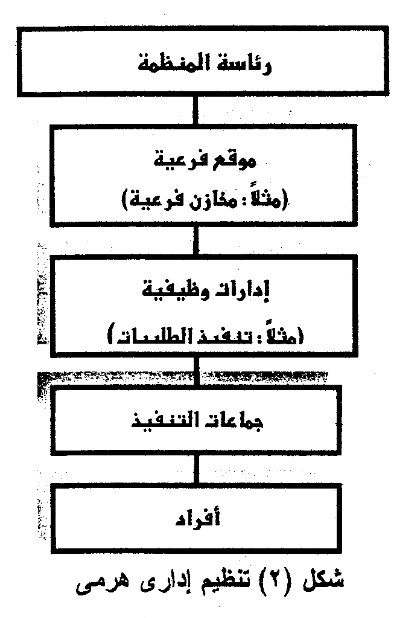
٤- **تقليل فرص الترقى الوظيفى للعاملين المعلوماتين** : إن وجـود مركـز حاسـبون مركزى ضخم يضم المتخصصون مـن المعلومـاتين فـى مكـان واحـد يعطـى فرصـاً أفضـل للعاملين للتطور الوظيفى عما لو وزعوا على مراكز محليـة أقـل حجمـا ومسـئوليات .

(٣) المستويات التنظيمية والنظم الموزعة :

ليس من المعتـاد أن توصـل كـل عقـدة Node [نقطـة مـن نقـاط الربـط فـى الشبكة] بكل عقدة أخرى ، كما أنه ليس مـن المعتـاد فـى حالـة توزيـع البيانـات أن يتـم ذلك على جميع العقد داخـل الشـبكة ، بـل الغـالب أن يعكـس الـهيكل التنظيمـى للشـبكة المعلوماتية الهيكل التنظيمى للمنشأة التـى يخدمـها .

ويبين شكل (٢) نموذج لهيكل تنظيمى ، وهـو هيكـل هرمـى تقليـدى يمثـل الكثـير من المنظمات الصناعية والخدمية . فالمنظمة لـها مقـر رئيسـى ، وعـدة فـروع محليـة ،

كأقسام الإنتاج ورقابة المخزون وتولى طلبيات البيع ، ويوجد فى كل قسم مجموعة عمل يقوم أفرادها بنفس المهام ، كالرد على إستفسارات العملاء فى قسم تولى طلبيات البيع . ويوجد أخيراً الأفراد الذين يمثلون أبسط " وحدة معالجة Processing unit " (الوحدة التى قد تحتاج دعما حاسوبياً) فى المنظمة .



شكل (٢) تنظيم إدارى هرمى

ومن إحتمالات وضع الهيكل بنائى للنظام المعلوماتى الموزع أن يكون توزيع البيانات وربط الأجهزة بنفس صورة الهيكل التنظيمى للمنظمة . فعلى سبيل المثال ، على مستوى مجموعات العمل ، ضمان توزيع الشبكة بين أفراد كل مجموعة . كما يتم الربط بين المستويات المختلفة بطبيعة الحال ، يتم بينها ما يلزم من تبادل للبيانات .

(٤) مدى التوزيع :

كان السؤال المثار فى الثمانيات هو عما إذا كان الأفضل الأخذ بنظام التوزيع أم الإستمرار فى الإعتماد على النظم الممركزة . أما فى أيامنا هذه فالسؤال يدور حول مدى تطبيق التوزيع ليمثل أقصى فائدة للمنشأة حالياً وفى المستقبل .

وهناك عدة عوامل تقنية تحكم مدى توزيع البيانات والأجهزة خاصة ما يتعلق منها بالإتصالات . ولكن بجوار ذلك فهناك إعتبارات أخرى على نفس القدر من الأهمية ، منها :

مستوى الوعى التعاونى لدى العاملين : إن إفتراض الإدارة حول السلوك الإنسانى له أثره فى درجة السيطرة الوظيفية عليهم . فهناك وجهة نظر ترى أن الموظف بطبيعته غير متحمس للعمل ويحتاج للرقابة المستمرة حتى يؤدى عمله بما يتفق وأهداف المنظمة . أما وجهة النظر المقابلة فترى أنه على العكس من ذلك ، متحمس بطبيعته للعمل ، ومستعد من تلقاء نفسه أن يبذل الجهد لتحقيق الأهداف التنظيمية . فإذا كانت النظرة الأولى هى السيطرة فلن يكون هناك ترحيب باللامركزية ، ويتمثل ذلك فى إنتهاج تسلسل هرمى مركزى ، وينعكس ذلك على النظم المعلوماتية فتميل نحو المركزية حيث يسهل تطبيق الرقابة . أما السياسة الإدارية الثانية فيغلب عليها إنتهاج الإستراتيجيات الإدارية التى تمنح الإستقلالية واللامركزية .

موقع صنع القرار : يرتبط بالنقطة السابقة تحديد من له حق صنع القرارات الهامة . وكلما إبتعدت مهمة صنع القرار عن المركزية فى المنشأة زاد إحتمال أن تكون الإمكانيات أيضاً لامركزية ، وهو الوضع المناسب لتطبيق النظم الموزعة .

الإعتماد المتبادل بين الأنشطة : إذا كان هناك إرتباط وثيق بين نوعين من الأنشطة فغالباً ما ستتم المعالجة الخاصة بهما فى مكان واحد ، فتوزيع المعالجة يجب ألا يحدث إلا بالنسبة للأنشطة غير المترابطة .

تجانس الأنشطة : أحياناً تكون الأنشطة مستقلة ، ولكن تدعو الحاجة لتخطيط مركزى للنظم المعلوماتية الموزعة ، مثال ذلك أن تمنح الإمتيازات للفروع ، ولكن التشابه التام فى طبيعة أنشطتها يجعل من المنطقى أن يكون لها نفس النظم المعلوماتية ، ويتم ذلك بأن يكون التخطيط لتلك النظم والرقابة عليها من خلال سلطة مركزية .

٥) توزيع البيانات :

قد يحدث في النظم الموزعة ألا تقتصر عملية التوزيع على المعالجة للبيانات المركزية التخزين بل يتم التوزيع للبيانات أيضاً . ويتم ذلك بعدة طرق تعكس مدى إستعداد المنشأة للتخلى عن الرقابة على البيانات ، بالإضافة للمواضيع التقنية الأخرى . ولعل أهم تلك الطرق مايلى :

١- يمكن إرسال نسخ من البيانات المركزية إلى الأفرع المحلية فى الشبكة كل يوم ، ويحدث ذلك نموذجياً حين تكون المسافات متباعدة بينها . ويتم التعامل مع هذه البيانات بصفة مستمرة مما يستلزم تحديثها ، وترسل تفاصيل هذه العمليات للموقع المركزى لتحديثها أولاً بأول ، أو ترسل كلها دفعة واحدة فى نهاية اليوم ، وفى كلتا الحالتين ترسل نسخ معدلة من البيانات الخزنة مركزيا صباح اليوم التالى لكافة الأفرع .

وتأثير ذلك الأسلوب يتمثل فى خفض تكاليف حركة المرور بين المواقع المحلية والمخزن المركزى للبيانات ، غير أن العيب فيه هو قدم المعلومات لمدة أربع وعشرين ساعة ، ويعتمد قبول ذلك أو رفضه على طبيعة النشاط .

٢- وفى صورة معدلة من النظام السابق ، وللتغلب على بعض مشاكل عدم الفورية ، يسمح للتغييرات المحلية بأن تحدث فى موقعها فور حدوث التغيير ، ثم ترسل البيانات المعدلة من كافة الموقع للمخزن المركزى فى نهاية اليوم ، ويرسل النسخة المعدلة من المخزون الكلى لكافة المواقع فى صبيحة اليوم التالى .

٣- لضمان الحصول على فورية البيانات وفى نفس الوقت توزيعها بحيث تعطى فوائد تخزين ومعالجة البيانات محلياً ، يجب إستخدام قاعدة بيانات موزعه بالكامل ، فيتم تخزين كل جزء منها فى موقع دون نسخ مكررة ، ويضمن برنامج إدارة قاعدة البيانات الموزعة وبرنامج إدارة شبكة الإتصالات أن تكون البيانات مرئية لكافة المواقع .

ويتزايد اليوم إستخدام قواعد البيانات الكاملة التوزيع ، ومن المحتمل أن تستمر هذه الموجه بالرغم من وجود مشكلات تقنية جديرة بالإعتبار بالنسبة لبرامج قواعد البيانات والإتصالات .

(٦) قواعد البيانات الموزعة وشبكات الإتصال :

Distributed data base & Commnication Network

١/٦ قواعد البيانات الموزعة :

إزداد إنتشار إستخدام الحاسبات الآلية فى أعمال الأقسام المختلفة للمنشآت خلال الخمسينيات والستينيات من القرن الميلادى الحالى . وكانت أعمال تلك الحاسبات تحت سيطرة وتحكم إدارة الأقسام المختلفة التى تستخدمها ، مما دعا إلى نمو لا مركزية معالجة البيانات . وقد أدى هذا الإتجاه فى النهاية – بعد وضوح عيوبه – إلى إتجاه مضاد لتحقيق مركزية معالجة البيانات . وقد إتخذ ذلك إتجاها من المنشآت إلى إقتناء الحاسبات الآلية الضخمة ذات الأجهزة والبرامج المتقدمة التى تحقق الإشتراك فى الإستخدام . وقد تأكد هذا الإتجاه أكثر وأكثر خلال الستينيات والسبعينيات .

وقد إستدعى ظهور الحاسبات متوسطة وصغيرة الحجم وتقدم وسائل الإتصالات إعادة النظر فى إتجاه التركيز السابق . فقد نشأ عن إستخدام الحاسبات الكبيرة عدة مشكلات وصلت فى بعض الحالات لإتخامها بالعمل المطلوب وعدم قدرتها على إنجاز المهام المحددة . بالإضافة إلى ذلك فقد إتضحت عدم قدرة أقسام معالجة البيانات المركزية على الإستجابة لكل إحتياجات المستخدمين بالسرعة المرجوة . وللتغلب على هذه المشكلة ، تم إستخدام إمكانات إتصالات البيانات لتطوير شبكات عمل مكونة من حاسب آلى مركزى كبير وعديد من الحاسبات الآلية الصغيرة أو المتوسطة أو حتى كبيرة الحجم (فى بعض الأحيان) التى جرى وضعها فى الأماكن التى تحتاج إلى إستخدامها بالأقسام المختلفة . وهو ما يطلق عليه مبدأ " المعالجة الموزعة للبيانات " التى يقل الإعتماد فى ظلها على الحاسب المركزى الكبير ويزيد الإعتماد على الحاسبات الصغيرة والمتوسطة اللامركزية الموزعة على أرجاء تنظيم المنشأة . فإذا إستخدم مبدأ لامركزية معالجة البيانات فإن ذلك يؤدى إلى توزيع طاقة معالجة البيانات حسب الإحتياج إليها فى كل من الأقسام .

وبناءا على ماتقدم نستطيع أن نستخلص بأن قواعد البيانات الموزعة يمكن تعريفها بأنها : " قاعدة بيانات متكاملة تبنى على قمة شبكة كمبيوتر بدلا من كمبيوتر واحد ، وتخزن البيانات التى تكون قاعدة البيانات فى مواقع Sites مختلفة

فى شبكة الكمبيوتر . كما أن برامج التطبيق التى تشغل بواسطة أجهزة الكمبيوتر يمكنها الوصول إلى البيانات فى مواقع مختلفة منها "

وبذلك فإن هذا المفهوم يركز على وجهتين أساسيتين لقاعدة البيانات الموزعة هما :

(١) التوزيع Distribution

توضح هذه الوجهة أن البيانات لاتتواجد فى نفس الموقع أو فى إطار المعالج Processor بل أنها تتواجد فى مواقع مختلفة تضمها عدة معالجات فى شبكة كمبيوتر.

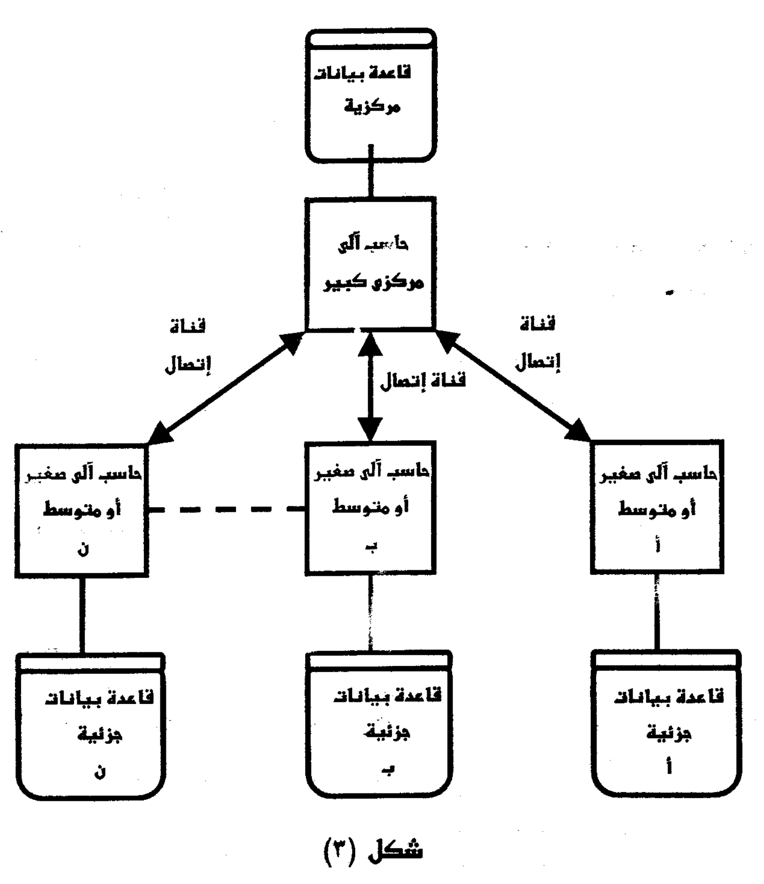
(٢) الترابط المنطقى : Logical Correlation

تبين هذه الخاصية بأن للبيانات بعض الخواص التى تربطها معا ، وبذلك تتميز قاعدة البيانات الموزعة عن قواعد البيانات أو الملفات الفرعية التى تتواجد فى مواقع مختلفة من شبكة الكمبيوتر وحتى يمكن توضيح هذا المفهوم السابق تحديده الذى يتكون من خواص التوزيع والترابط المنطقى ، نستعرض فيما يلى مفهوم قواعد البيانات الموزعة بطريقة أكثر وضوحا .

ويمكن لنموذج قاعدة البيانات الموزعة من أن يتواءم مع التنظيمات المركزية لكثير من المنظمات المعاصرة ، حيث يمكن فى بعض النظم الإحتفاظ بالبيانات فى الحاسب الآلى المركزى ، وتعطى الحاسبات الآلية الصغيرة والمتوسطة اللامركزية صلاحية طلب هذه البيانات من الحاسب المركزى (أنظر الشكل ٣) . وبالرغم من إتاحة هذا الأسلوب للتحكم المركزى فى البيانات ، فإن ذلك يكون مكلفا ولا يحقق الكفاءة ، لإنشغال قنوات الإتصال دائما وبصفة مستمرة لتوفير البيانات للحاسبات اللامركزية .

شكل رقم (٣) يوضح الأسلوب الأول لشبكات العمل الموزعة لقواعد البيانات

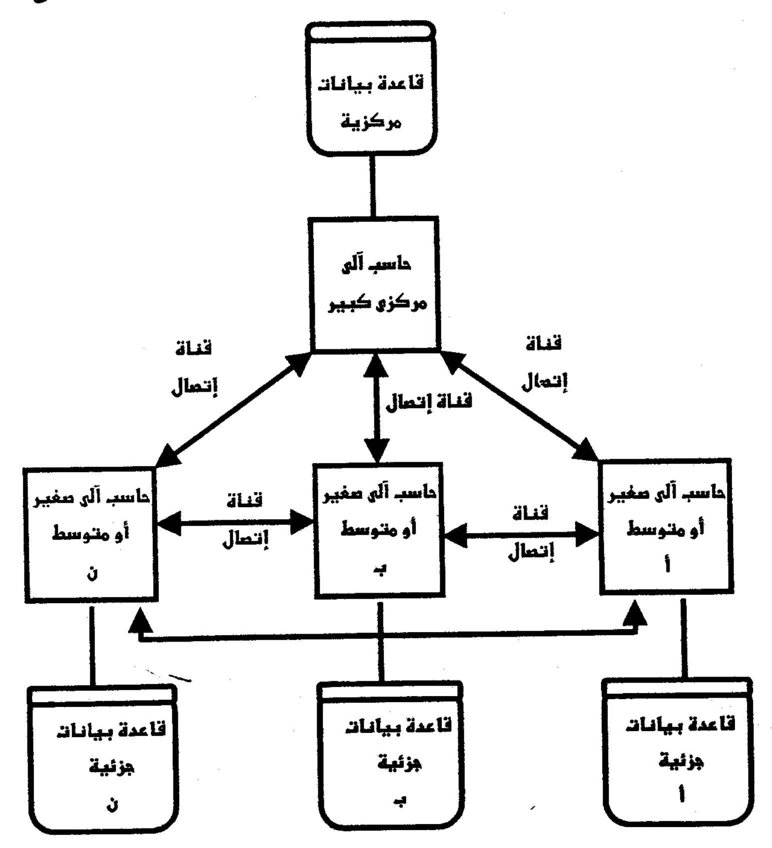
ويحقق الأسلوب الثانى نوعا من الإشتراك فى النظام . فيتم فيه الإحتفاظ بأغلب البيانات فى قاعدة البيانات المركزية بالحاسب المركزى الكبير فى الوقت الذى تنسخ فيه صور لبعض هذه البيانات ليحتفظ بها فى الحاسبات الآلية الصغيرة أو المتوسطة الموزعة على أرجاء التنظيم . وتمثل النسخ المختلفة لقاعدة البيانات المركزية بالقواعد الفرعية للبيانات المطلوبة لمواجهة إحتياجات الأقسام المختلفة من هذه البيانات كل فى حدود إطار أعماله وأنشطته . ويتيح هذا الأسلوب – كما هو موضح فى الشكل (٤) – طريقة أسرع وأقل تكلفة للوصول للبيانات المطلوبة عن الأسلوب الأول السابق .



شكل (٣)

الأسلوب الثانى لشبكات العمل الموزعة لقواعد البيانات

ويسمح الأسلوب الثالث لكل حاسب آلى صغيرة أو متوسط فرعى بأن يحتوى على قاعدة بيانات فرعية للقيام بعمليات القسم المختص (أنظر الشكل ٤) . ولا تمثل قواعد البيانات الفرعية نسخا جزئية مكررة من قاعدة البيانات المركزية فى هذا



شكل (٤) الأسلوب الثالث لشبكات العمل الموزعه لقواعد البيانات .

الأسلوب . وتنحصر الصعوبة الأساسية فى هذا الأسلوب فى ضرورة توفير إمكانية تبادل البيانات بين الحاسبات الآلية الصغيرة أو المتوسطة الفرعية وبعضها ، كل طبقا لإحتياجاته ، وكذلك بينها وبين القاعدة المركزية للبيانات ضمن إطار الحاسب الآلى المركزى ، وهذا مما يتطلب توفير أنواع معقدة من البرامج التى تقوم بذلك .

٢/٦ نظم إدارة قواعد البيانات الموزعة

تساند نظم إدارة قواعد البيانات الموزعة (DDBMS) عملية إنشاء وصيانة قواعد البيانات الموزعة . وقد طور البعض منها من نظم إدارة قواعد البيانات المركزية التى توفر تجاريا من قبل الشركات المطورة لها . وتتضمن برامج هذه النظم عدة مكونات تتمثل فيما يلى :

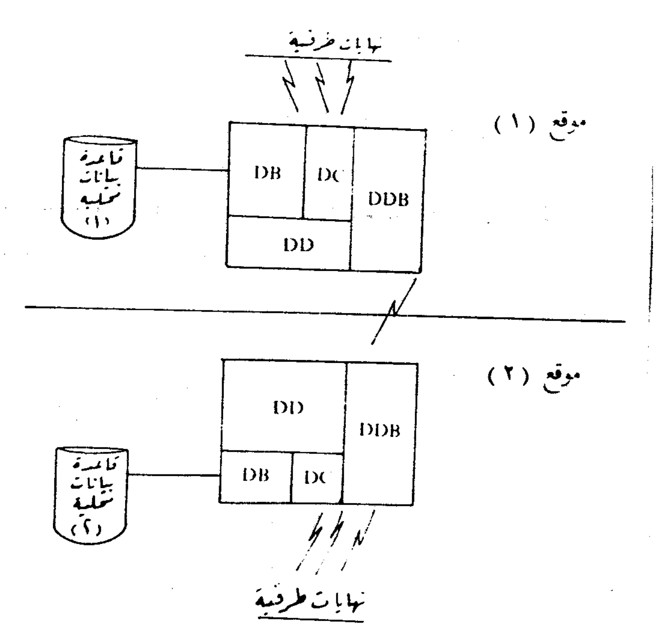
١-إدارة قاعدة البيانات .

٢-إتصال البيانات .

٣-قاموس المصطلحات يمتد لعرض البيانات فى الشبكة (DD)

٤-قاعدة البيانات الموزعة (DDB) .

ويتضح ذلك فى الشكل التالى رقم (٥) :



شكل رقم (٥) يوضح مكونات نظم إدارة قواعد البيانات الموزعة

يتضح من هذا الشكل أن نظم إدارة قواعد البيانات الموزعة تتضمن على أربعة مكونات رئيسية يكون من بينها مكون قاعدة البيانات الموزعة (DDB) ، علما بأن نظام إدارة قواعد البيانات المركزية العادية يتكون من المكونات الثلاثة الأخرى وهى مكون قاموس البيانات (DD) ومكون إتصال البيانات (DC) ، ومكون قاعدة البيانات (DB) ، أما الخدمات التى تساند هذا النوع من النظم فيمكن أن تحدد فيما يلى :

١ - الوصول المباشر لقواعد البيانات من بعد عن طريق برنامج تطبيق خاص ، وتعتبر هذه الخاصية من أهم خواص قواعد البيانات الموزعة .

٢ - توفر درجة معينة من شفافية التوزيع حيث تساند هذه الخاصية النظم بأساليب مختلفة عن طريق المقايضة بين برامج الشفافية وآداء التوزيع .

٣ - مساندة إدارة وضبط قاعدة البيانات التى تشتمل على أدوات لمراجعة قاعدة البيانات ، وتجميع المعلومات عن مدى الاستفادة وتوفير نظرة شمولية للملفات .

٤ - بعض أساليب المساندة للضبط المتزامن وإستعادة التصرفات أو الأفعال الموزعة ...الخ .

٣/٦ شبكات الكمبيوتر

يتوفر فى شبكات الكمبيوتر Computer Networks عدة مواقع فى كل منها كمبيوتر مضيف Host computer ترتبط معا بواسطة شبكة اتصال تشتمل على قنوات اتصال عديدة مثل خطوط التليفون والكابلات المحورية Coaxial cables والأقمار الصناعية Satellite .... الخ بالاضافة الى أجهزة الكمبيوتر المتعددة . أما أجهزة الكمبيوتر التى لاتكرس لوظيفة الاتصال فإنها لاتعتبر أجهزة كمبيوتر مضيفة .

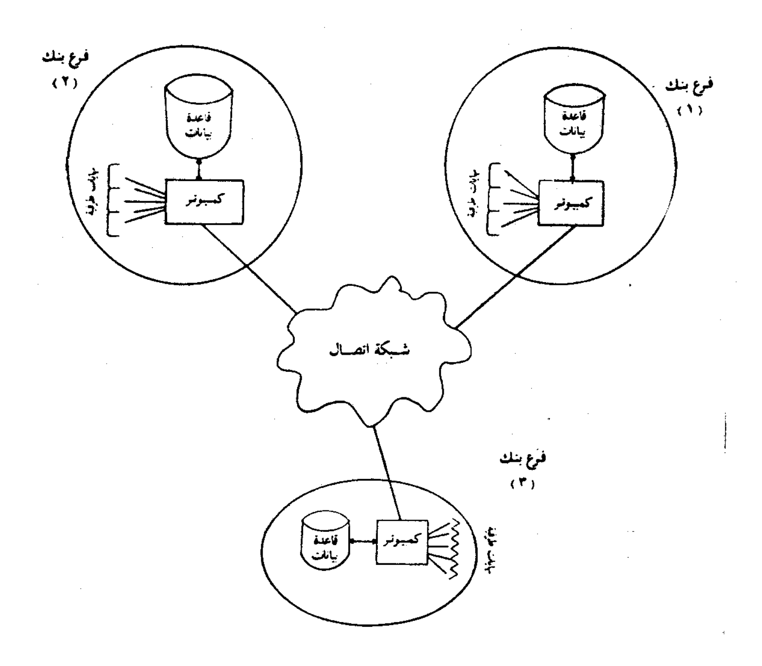
والتسهيلات الأساسية التى توفر بواسطة شبكة الكمبيوتر تتمثل فيما يلى :

١- العملية أو التطبيق الذى يؤدى فى أى موقع من مواقع الشبكة.

٢- إرسال إشارة لعملية تشغيل فى موقع آخر من الشبكة .

٣-شبكة الاتصال المستقلة عن مواقع الشبكة .

ويمثل الشكل التالى رقم (٦) نموذج شبكة الكمبيوتر :



شكل رقم (٦) يوضح نمـــوذج شـــبكة كمبيوتـــر

ولأى شبكة كمبيوتر عدة أبعاد تتصل بوظيفة الاتصال وتتحـــدد فيمـــا يلـــى :

١-التأخير Dealy الذى يرتبط بتوصيل الرســـالة الـــى وجهتـــها عندمـــا يكـــون اســـتخدام الشبكة فى أقل معدلاته ، ويعتمد التأخير علـــى خصـــائص مكونـــات شـــبكة كمبيوتـــر ، وإذا طال الانتظار فإن التأخير يطول أيضا مما يتطلـــب تحليـــل صفـــوف الانتظـــار حتـــى يقوم التأخير فى عملية التوصيـــل .

٢-تكلفة توصيل الرسالة حيث ان لكل رسالة تكلفة ثابتـــة بالاضافــة الــى تكلفــة متغــيرة تعتمد على طول الرســـالة .

٣-مدى الوثوق من الشبكة Reliability فتوصل الرســالة لوجهتــها بدرجــة علايــة مــن الصحة والوثوق إذا كانت عملية البث فى المواقع المختلفــة موثــوق منــها آليــا.

ولشبكات الكمبيوتر أنواع وأشكال عديدة فمنـــها مـــاهو محلـــى ومنـــها مـــاهو واســـع المدى ، ومنها ماهو إقليمى ، كما تأخذ الشبكة عـــدة أشـــكال فـــى توصيلـــها فمنـــها الشـــبكة النجمية ، والشبكة الحلقية ، وشبكة الجســـر ، والشـــبكة المزعـــة . وليـــس مجالنـــا هنـــا أن نتعرض بالتفصيل لتلك الأشكال ولكن أكتفينا فقـــط بـــالتركيز علـــى النظـــم الموزعـــة عمومـــا وقواعد البيانات الموزعة مع التعرض بشكل سريع لمفـــهوم وفكـــرة شـــبكات الحاســـب .

الفصل السابع

تخطيط قواعد البيانات

الفصل السابع
تخطيط وتصميم قواعد البيانات

مقدمة:

يعتبر تخطيط قاعدة البيانات من الأساليب الجوهرية عندما ترغب المنظمة فى جنى ثمار ومزايا إستخدام إدارة موارد البيانات بها ، وبذلك يجب أن تساند الخطة التى تعد لقاعدة البيانات الخطة الشمولية لنظام المعلومات المتكامل بالمنظمة الذى يحقق أهدافها.

وسوف نستعرض من خلال هذا الفصل المنهج الذى يجب أن يتبع عندما نبدأ فى تخطيط قاعدة البيانات على مستوى المنظمة ، ويستخدم هذا المنهج المدخل التخطيطى الذى يبدأ من أعلى لأسفل أى من الإدارة العليا إلى الإدارة الإشرافية بالمنظمة ويتمثل أساس أى جهد تخطيطى فى إطار خطة العمل على تحديد الأهداف والسياسات والقيود التى تحيط بالمنظمة ، كما تعرف مجموعة الوظائف والعمليات والأنشطة والأحداث التى يجب أن تؤديها هذه المنظمة . على أن الهدف الرئيسى لتخطيط قاعدة البيانات يتمثل فى تطوير نموذج فكرى للبيانات قد يطلق عليه " نموذج المنظمة " يوضح العلاقات بين الكيانات المختلفة المهمة المتواجدة فى المنظمة . وتشتمل خطة قاعدة البيانات على إعداد جدول شامل لتصميم وتنفيذ قاعدة البيانات . وبذلك يوضح هذا الإستعراض عملية تخطيط قاعدة البيانات وتوضيح ذلك ببعض الحالات الواقعية لقاعدة البيانات المخططة .

(١) أهمية تخطيط قواعد البيانات:

من الملاحظ أن نظم المعلومات فى المنظمات المختلفة لم تحظى بقدر كاف من التخطيط والتصميم المتأنى ، فى الوقت الذى تتطور فيه على نسق واحد تقريباً يبدأ من أسفل لأعلى ، وتغلب عليها الصفة الإستقلالية لكل نظام ، وتعد لحل مشاكل

تطبيقية متفرقة . هذه النظـــم التقليديـــة للمعلومـــات تتصـــل فـــى العـــادة بـــالإجراءات أو برامج التطبيق المطلوبة لحل المشاكل التى تتواجـــد فـــى الواقـــع .

هذا المدخل التقليدى فى تطور نظـــم المعلومـــات يتضمـــن مشـــاكل عديـــدة تتمثـــل فى أن الإجراءات التنظيميـــة المطلوبـــة حاليـــاً يحتمـــل أن تتغـــير وتتبـــدل علـــى مـــرور الزمن تبعاً للمتغيرات البيئية التى تحدث تباعـــاً . فعلـــى ســـبيل المثـــال قـــد تغـــير إحـــدى الشركات إرسال الفواتير لعملائـــــها أو قـــد تغـــير الجامعـــة أو المعـــهد العلمـــى إجـــراءات تسجيل الطلاب فيها ، إلا أنه عند حـــدوث هـــذه المتغـــيرات ففـــى العـــادة تعديـــل برامـــج التطبيق القائمة بـــالفعل .

وفى مواجهـــة هـــذا المدخـــل التقليـــدى ، نجـــد أن إدارة مـــوارد البيانـــات تتصـــل بصفه مستمرة وأساسية بمتطلبات قـــاعدة البيانـــات المطلـــوب تطويرهـــا للمنظمـــة لكـــى تلبـــى حاجاتـــها للمعلومـــات فـــى الحـــاضر والمســـتقبل . وعنـــد الأخـــذ بـــهذا المدخـــل التخطيطى المتطور تتواجد مزايا رئيسية تعود على المنظمة بـــالنفع إلـــى حـــد كبـــير .

ومن المزايا الرئيسية لهذا المدخـــــل هـــو أن بيانـــات الشـــركة أو المنظمـــة تعتـــبر أقل إحتمالاً للتغيير أو أنها سوف تتغير بمعدل أقل مـــن تعديـــل الإجـــراءات . فعلـــى ســـبيل المثال ، أن لم تغير المنظمة من آدائها وأعمالها تغيـــيراً جوهريـــاً ، فـــإن هيكـــل البيانـــات الأساسية سوف يبقى ثابتاً خلال فـــترة طويلـــة نســـبياً إلـــى حـــد مـــا ، بـــالرغم مـــن أن الإجراءات التى تستخدم للوصول للبيانات ومعالجتها ســـوف تتغـــير مـــرات عديـــدة خـــلال تلك الفـــترة .

من هذا المنطلق فـــإن التحـــدى الـــذى يواجـــه المنظمـــات المعـــاصرة يتمثـــل فـــى تخطيط قواعـــد بيانـــات ثابتـــة بحيـــث تكـــون مســـتقلة نســـبياً عـــن اللغـــات والـــبرامج المستخدمة فى تحديث هـــــذه البيانـــات . وحتـــى يمكـــن للمنظمـــة مـــن الإســـتفادة بـــهذا المدخل الحديث ، فإنها تقوم بتحليل حاجاتـــها مـــن المعلومـــات وتخطـــط قـــاعدة بياناتـــها بعناية فائقـــة .

وعند محاولـــة إدخـــال أســـلوب قـــاعدة البيانـــات فـــى المنظمـــة بـــدون التخطيـــط المسبق لها ، فإن ما يتحقق من نتائج تبعـــاً لذلـــك يصبـــح هامشـــياً أو عديـــم الجـــدوى ، كما يثير عديد من المشاكل بدلاً من حلـــها . وقـــد تســـهم قـــاعدة البيانـــات فـــى مســـاندة

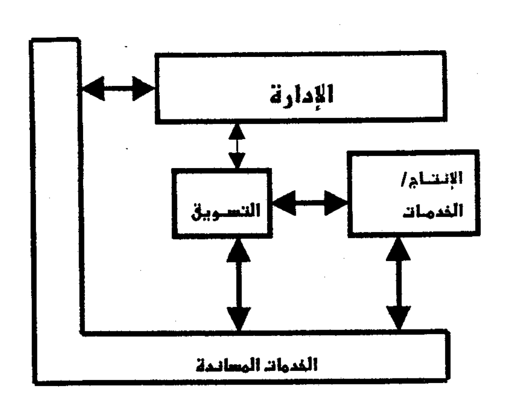
التطبيقات الفردية ولكنها لاتقدر على توفير مورداً أساسياً يشارك فى تقدم المنظمة ذاتها . كما تصبح نظم إدارة قواعد البيانات الجاهزة طريقة مكلفة فى الوصول لبيانات المنظمة حيث توفر نفس التقارير الهيكلية والنظم التى توفرها التطبيقات الفردية السابقة ، كما تستمر مشاكل صيانة البرامج محددة ومقيدة لنظم معالجة البيانات التقليدية .

إلا أنه فى إطار المدخل التخطيطى المتطور الذى يرتكز على البيانات كمورد تنظيمى ذا قيمة كبيرة للمنظمة يصبح لعملية تخطيط قاعدة البيانات الدور الجوهرى فى الإستفادة القصوى بقواعد البيانات . وبذلك يجب على المنظمة من أن تطور خطط إستراتيجية لإدارة بياناتها كما يحدث عندما تخطط لتنمية مواردها البشرية والمالية والمادية الأخرى . وعملية التخطيط يجب أن تتم على أساس تنازلى من أعلى لأسفل حتى يمكن أن تتكامل نظم المعلومات وخطط قواعد البيانات مع أهداف المنظمة الرئيسية والأساسية . وسوف يتمخض ذلك فى بزوغ خطة شمولية لقاعدة البيانات توفر خريطة مسار تسهم فى تصميم وتنفيذ عمليات الإستفادة من قواعد البيانات فى المنظمة .

وعلى الرغم من عدم توفر منهج معين أو مدخل معيارى مقنن يستخدم فى تخطيط قواعد البيانات بصفة عامة ، إلا أننا سوف نستعرض مجموعة من الأساليب تحدد بعض الأسس التى إستخدمتها بعض المنظمات عند إدخال قواعد البيانات بها .

(٢) **البيئة التنظيمية لتخطيط قواعد البيانات** :

الهدف الرئيسى من تخطيط قاعدة البيانات هو تطوير " نموذج بيانات إستراتيجى Strategic data model " الذى يمثل إحتياجات المنظمة من المعلومات فى الحاضر والمستقبل . كما يجب أن يكون هذا النموذج مفهوماً ومسانداً من قبل المستويات الإدارية المختلفة فى المنظمة وما تؤدية من وظائف عديدة يمكن أن تتحدد مجالاتها الأساسية فى الشكل التالى :



شكل رقم (١) مجالات الوظائف الأساسية فى المنظمة

يتضح من الشكل السابق أن هناك أربعة مجالات أساسية للوظائف التى تتواجد فى معظم المنظمات وهى وظائف الإنتاج والتسويق والإدارة والخدمات المساندة التى تتفاعل بعضها مع بعض .

ويمثل مجال الإنتاج أو الخدمات الجانب الهام لتواجد المنظمة ويرتبط ذلك فى تزويد المنتجات أو الخدمات لبيئة المنظمة الممثلة فى العملاء التى تتعامل معهم . وقد يشتمل هذا المجال الوظيفى الهام على وظائف فرعية مثل إدارة المواد ، وجدولة الإنتاج والرقابة على المنتجات ، والإنتاج .. إلخ .

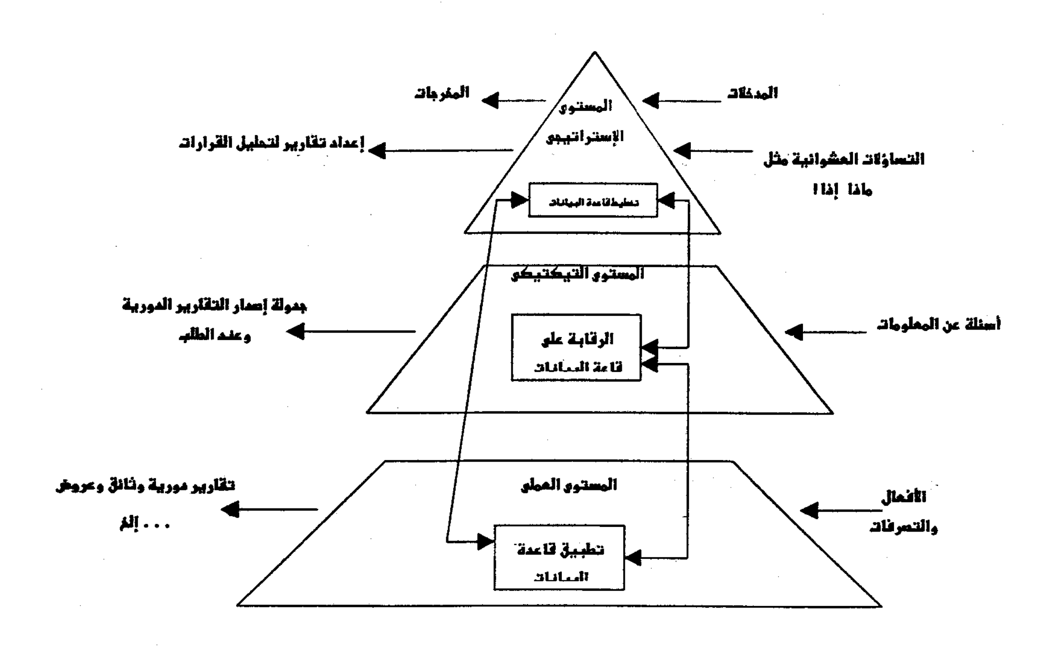
أما مجال التسويق فيعبر عن الوظيفة المسئولة بتحديد الطلب على المنتجات أو الخدمات وتعريفها والترويج لها . وقد تتفرع وظيفة التسويق الرئيسية إلى عدة وظائف فرعية كالمبيعات ، وبحوث السوق ، والتوزيع والدعاية والإعلان .. إلخ .

ويشتمل مجال الخدمات المساندة عن وظائف فرعية كالهندسة والمحاسبة والأفراد والصيانة .. إلخ .

بينما يكون مجال الوظيفة الإدارية الأساسية فى المنظمة مرتبطاً بالتخطيط والتوجيه والرقابة والمتابعة فى المنظمة .

وعادة يصعب على أى منظمة من المنظمات تنفيذ قاعدة بياناتها مرة واحدة أو فى مدى زمنى قصير . وبذلك تعمل على التخطيط لإدخال قاعدة البيانات خلال مراحل متعددة وعلى مدى زمنى طويل نسبياً قد يمتد لعدة سنوات حيث أنه ينتج

عن ذلك تطبيق لقاعدة بيانات واحـدة وكبـيرة إلـى حـد مـا أو تطبيقـات لعـدة قواعـد بيانات تخدم نظام المعلومات المتكامل بالمنظمـــة . ولــهذا السـبب فمـن الهمـه جـداً أن يكون للمنظمة التى تحاول إدخال قاعدة البيانـات كأسـاس لنظـام المعلومـات بـها خطـة إستراتيجية ونموذج بيانات لتوجيه الجهود المختلفة نحو التنفيـذ حيـث أنـه بمعـزل عـن هذه الخطة فإن تطوير قاعدة البيانات سوف يكـون علـى أسـاس غـير متنـاسـق وبذلـك تزداد إحتمالات القصور والمشاكل التى تصحـب هـذا التنفيـذ بالإضافـة إلـى المجـالات الرئيسية للوظائف وحاجتـها إلـى تخطيـط قـاعدة بيانـات لكـى تخدمـها فـإن البيئـة التنظيميـة للمنظمـات المعـاصرة تشـتمل علـى ثـلاث مسـتويات لـلإدارة والتخطيـط والرقابة تتمثل فى المستويات الإسـتراتيجية والتكتيكيـة والعمليـة أو التطبيقيـة. هـذه المستويات الإدارية الثلاثة يمكن توضيحها فـى الشـكل التـالى :



شكل رقم (٢) المستويات الإدارية والحاجة للمعلومات

ويلاحظ فى الشكل السابق أن التخطيط الإستراتيجى يمثل عملية تقرير أهداف المنظمة وتحديد الموارد المطلوبة لتحقيق الأهداف ورسم السياسات المشكلة لقواعد التزود بالموارد وإستخدامها . ويعتبر مديروا الإدارة العليا الإستراتيجية مسئولين عن الآداء الشمولى والمتكامل للمنظمة . وكما هو مبين فى الشكل السابق فإن الإدارة الإستراتيجية للمنظمة تعتمد فى آداء مهامها وإتخاذ قرارتها على معلومات ومعارف مختصرة إلى حد كبير . فعلى سبيل المثال سوف يختص رئيس المنظمة بالعائد الذى سيعود على المنظمة من الإستثمار أو حصى المنظمة فى السوق المحلى او الخارجى وهكذا . بدلاً من شغل وقته بآداء معين أو تشغيل خط إنتاج ما . وبذلك فإن الحاجة فى هذا المستوى تكون على أساس توفير نظام مساندة القرار Decision (DSS) Suppert System الذى يجيب على التساؤلات العشوائية للمعلومات ويخرج عدة أساليب محاكاة لبدائل تخطيط متعددة يحتاجها الرئيس أو المدير المسئول . كما أن رجال الإدارة العليا الإستراتيجية يحتاجون عادة إلى قدر كبير من المعلومات الخارجية المتصلة بالتنبؤات ومعلومات عن الأوجة الإقتصادية والمنافسين وما شابه ذلك .

أما المستوى التيكيتكى من الإدارة فإنه يختص بالرقابة الإدارية والمتابعة وفى هذا المستوى الإدارى يتحقق المديرون ويتأكدون من أن الموارد أمكن الحصول عليها وإستخدمت بكفاءة وفعالية لتحقيق الأهداف المختلفة . ويستخدم فى ذلك أساليب موزانة إستخدام الموارد ، وقياس التقدم فى مواجهة الخطط المعدة سلفاً ، وتصحيح الأفعال والتصرفات عندما تستدعى الحاجة . وكما هو مبين فى الشكل السابق أيضاً فإن المديرين فى هذا المستوى التيكتيكى يتطلبون معلومات تجميعية تلخص البيانات التشغيليه أو التطبيقية التى تعرض فى تقارير رقابة وتقارير عن الميزانية . وذلك يمكن لهذه المعلومات أن تجيب عن التساؤلات العرضية كما تعرض ايضاً فى شكل تقارير دورية .

أما المستوى التطبيقى أو التشغيل لإدارة المنظمة فإنه يرتكز على تنفيذ المهام والأنشطة المختلفة بالمنظمة وبذلك فإنه يختص بجدولة المهام الفردية والرقابة عليها وأداء التصرفات الفردية المعينة .

أما البيانات التى تطلب لآداء هذا المستوى التشغيلى فهى بيانات تفصيلية غير نقدية فى الغالب تبنى على التصرفات المختلفة كما فى حالة تسجيل المرضى أو الطلاب وإستلام الموارد وتكاليف الأفراد .

وتتمثل البيانات التى تنتج من هذا المستوى التشغيلى فى الوثائق والعروض التطبيقية المفصلة التى تفيد فى عملية إتخاذ القرارات التشغيليه إلى حد كبير .

تلك هى البيئة التنظيمية لقاعدة البيانات التى يجب أن تحدد ويتعرف عليها وتعد نماذج لإحتياجات المستويات المختلفة والوظائف المتنوعة من المعلومات ودورة سريانها فى إطار المنظمة حتى يمكن التخطيط السليم لقاعدة البيانات ..

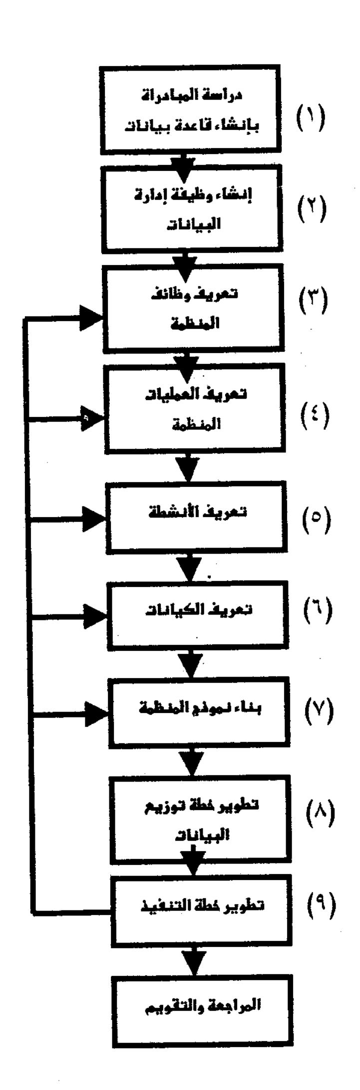
وبذلك فإن الغرض من تخطيط قاعدة البيانات ، هو تطوير خطة إستراتيجية طويلة الأجل لبيئة قاعدة البيانات التى تساند إحتياجات المنظمة للمعلومات فى الحاضر ولما هو مخطط له مستقبلاً . وتمثل خطة قاعدة البيانات خطة فرعية لتخطيط نظام المعلومات المتكامل فى المنظمة الذى يمثل فى حد ذاته خطة فرعية من الخطة الشمولية لتخطيط المنظمة . ومن هذا المنطلق تصبح عملية التخطيط فى المنظمة ذات طبيعة هرمية من أعلى لأسفل . أما المدخل البديل لذلك فهو ما يكون فيه التخطيط من أسفل لأعلى الذى قد يفيد فى إستقراء النظم وحاجات المعلومات القائمة بالفعل من واقع المهام الفردية التفصيلية إلا أن ذلك قد لا يسهم فى تلبية حاجات المعلومات للمستويات الإدارية الإستراتيجية .

ويعتبر تخطيط قاعدة البيانات مشروعاً ذا صيغة رسمية وذا صفة طويلة الأجل . وبذلك يخصص له فريق عمل مكون من مجموعة من الأفراد الأخصائيين تتفاوت تخصصاتهم وأعدادهم طبقاً لحجم العمل ودرجة التعقيد التى تتواجد فى المنظمة .

وسوف نتعرض فى المناقشة التالية للمهام التى تتضمن فى تخطيط قاعدة البيانات ، والتخطيط الهرمى الذى يواجه التصميم التصاعدى والمنهج الممكن إتباعه فى التخطيط بالإضافة إلى حالة فعلية لتخطيط قاعدة بيانات فى إحدى المستشفيات .

٣) المهام المتضمنة فى تخطيط قاعدة البيانات :

يمكن التعرف على المهام التى تتضمن فى عملية تخطيط البيئة التى ستتواجد فيها قاعدة البيانات من خلال إستعراض الشكل التالى :



شكل رقم (٣)

يلاحظ على يسار الشكل السابق أسهم التغذيـــة العكسـية التـى تبيـن أن تخطيـط قاعدة البيانات ليس عملية فحســب . حيـث أن عمليـة التخطيـط تشـتمل علـى بعـض المهام التى فى حاجة مستمرة إلى تحسين وتعديل وتعزيـز وخاصـة عندمـا يكـون هنـاك معرفة كافية بمتطلبات المنظمة من البيانات . وفـى الإسـتعراض التـالى سـوف نناقش بإختصار المهام الرئيسية المتضمنة فـى تخطيـط قـاعدة البيانـات والتـى وضحـت فـى الشكل الســابق :

١ - دراسة المبـادأة بإنشـاء قـاعدة البيانـات :

تعرف هذه المهمة الأولي من مهام تخطيط قـاعدة البيانـات أهـداف مدخـل قـاعدة البيانات في المنظمة ، وبذلك يجب دراسة وتعريـف المشـاكل التـي تتواجـد فـي بيئـة المنظمة الحالية ، وتحديد المزايا التـي سـتعود علـي المنظمـة نتيجـة إدخـال قـاعدة البيانات . وتسجل هذه المزايا المتوقعة في شـكل أهـداف تتفـق مـع الخطـط الطويلـة الأجل في المنظمـة .

٢ - إنشـاء وظيفـة إداري أو مديـر البيانـات :

تفسر فـي هـذه المهمـة وظـائف ومسـتويات إدارة البيانـات ومواصفـات "إداري قاعدة البيانات" الذي يكلف بدور قيادي في عمليـات التخطيـط اللاحقـة ومـدي تبعيتـه لإدارة المنظمة وإدارة معالجة البيانات بـها وتحديـد الأدوات الضروريـة التـي يحتـاج إليها ويعمـل علـي تطويرهـا مثـل قـاموس البيانـات Data Dictionary الـذي يساند كل أو معظم المهام اللاحقة في عمليـة التخطيـط .

٣ - تعريـف وتحليـل نظـم العمـل بالمنظمـة :

يحدد في هذا النشاط أربـعة مهام أساسية كمـا هـو مـدرج فـي الشـكل السـابق. حيث يقوم فريق العمل بتحليل وتعريف وتوثيـق وظـائف وعمليـات وأنشـطة وكيانـات المنظمـة .

٤ - بنـاء نمـوذج المنظمـة :

يمثل بناء المنظمة المهمة الأساسية المركزيـة فـي عمليـة التخطيـط الاسـتراتيجي للمنظمة ككل . ويوضح هذا النموذج الكيانـات الرئيسـية التـي يشـتمل عليـها التنظيـم

ويحدد العلاقات والارتباطات المختلفة التي قد تتواجد بين هذه الكيانات بعضها ببعض .

٥- **تطوير خطة توزيع البيانات :**

في هذه الخطوة تحدد أساليب بث وتوزيع البيانات من قواعد البيانات الطبيعية للمنظمة . كما تعرف قنوات البث وأساليب عرض البيانات ودوريتها ودرجة الاختصار وفئات المستخدمين .... إلخ .

٦- **تطوير خطة التنفيذ :**

يعد في هذه المهمة جدولة وتوقيت ومسئولية تنفيذ خطوات وأنشطة قاعدة البيانات . كما تنشأ في هذه المهمة أولويات مختلفة لربط قاعدة البيانات الاستراتيجية مع خطط نظم المعلومات المتكاملة في المنظمة .

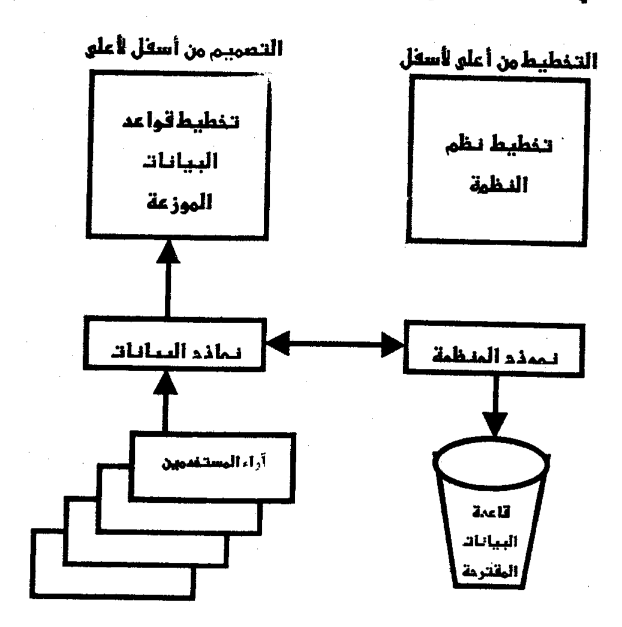
٧- **المراجعة والتقويم:**

يعد في هذه المهمة تقريرا نهائيا يراجع كل مكونات عملية التخطيط حتى يمكن التأكد من أنها متماسكة ومترابطة ويحدد فيه أساليب المتابعة والتقويم المستمرة وطرق تحديث البيانات التي تشتمل عليها ملفات قاعدة البيانات .

(٤) التخطيط من أعلى لأسفل والتصميم التصاعدي من أسفل لأعلى :

تتطلب عملية تطوير قاعدة البيانات استخدام أساليب التخطيط التنازلية أو الهرمية من أعلي لأسفل والتصميم التصاعدي من أسفل لأعلي . وتبدأ عملية التخطيط من أعلي لأسفل بتحديد الأهداف التنظيمية الأساسية ويستخدم في ذلك تخطيط النظم عن طريق تحليل الوظائف والعمليات والأنشطة والكيانات التنظيمية التي سبق عرضها في مهام التخطيط . ثم يطور بعدئذ نموذج المنظمة الذي يمثل رسما موضحا به كيانات التنظيم الرئيسية ومدي الترابط بينها . وقد يطلق علي هذا النموذج خريطة الكيان Entity Chart وهو لا يوضح نموذج تفصيلي للبيانات بل يمثل شكلا تخطيطيا لهذه الكيانات . وعن طريق تحليل نموذج المنظمة يمكن تقسيم الجهد الإجمالي في تطوير أو استخدام قاعدة البيانات أو عدة قواعد بيانات تقترح

للمنظمة. أما تصميم قاعدة البيانات بتعمق وتفصيل أكبر فيتم من أسفل لأعلي ويوضح ذلك في الشكل التالي :



شكل رقم (٤)

يوضح التخطيط من أعلى لأسفل فى مواجهة التصميم من أسفل لأعلى

وفي تصميم قاعدة البيانات كما يبينه الشكل السابق يبدأ المحللون باستقصاء أراء المستخدمين للبيانات ويطبقوا في هذا الصدد بعض أساليب المعايرة والتقنين في إطار تطوير نماذج بيانات مفصلة التي تراجع علي نماذج المنظمة لكي يتأكد من اكتمالها ودقتها ، ومن هذا المنطلق تقوم نماذج المنظمة بآداء أدوار مهمة في تخطيط قاعدة البيانات من أعلي لأسفل تتمثل فيما يلي :

١-توفير الأساس المرجعي الشمولي والمتكامل لكيانات وبيانات المنظمة.

٢-تقديم القاعدة العملية التي عن طريقها يمكن تقسيم نموذج بيانات المنظمة إلي مراحل وملفات البيانات الممكن إدارتها .

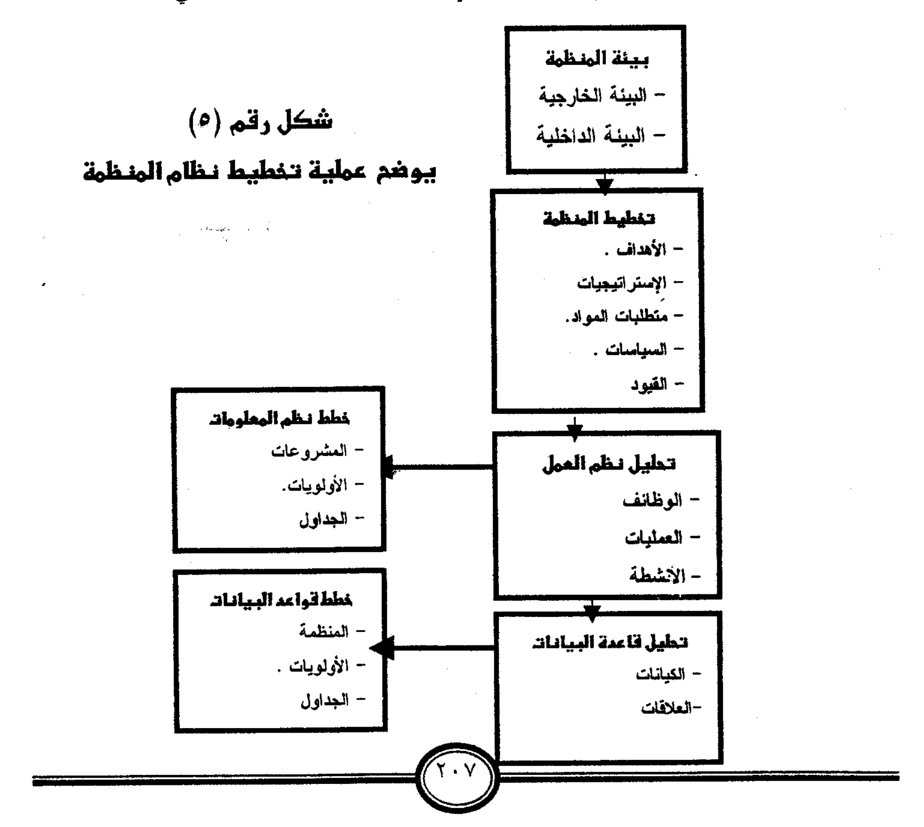
٣-تسهم في مراجعة أساليب تصميم قاعدة البيانات التفصيلية من الوجهة الشمولية المتكاملة للمنظمة .

(٥) منهج تخطيط قاعدة البيانات:

يبني منهج تخطيط قاعدة البيانات من أعلي لأسفل علي مدخل تخطيط نظم العمل التي تساعد المنظمة في إنشاء خطة تضم كل مكونات النظام التي تفترض أن خطة نظام المعلومات بالمنظمة يجب أن تكون متكاملة مع خطة المنظمة ، ويجب أن يطور ذلك من وجهة نظر الإدارة العليا بالمنظمة وبمشاركتها الفعلية والنشطة في هذا التطوير .

ويجمع المنهج المتبع بعض خصائص تخطيط النظم بالمنظمة مع مداخل تخطيط قاعدة البيانات المتصلة بهذه المنظمة ، وتعتبر عملية تخطيط قاعدة البيانات جزءا من تخطيط النظام الشمولي بالمنظمة . وفي هذا الإطار يمكن تحديد أربعة أوجه رئيسية لعملية تخطيط نظام المنظمة والتي يتضمنها الشكل التالي :

شكل رقم (٥)
يوضح عملية تخطيط نظام المنظمة



١- تعريف بيئة المنظمة :

يتضمن تعريف بيئة المنظمة تحديد كل من البيئة الداخلية والخارجية التي تتواجد فيها المنظمة . وتشتمل البيئة الخارجية للمنظمة على كل من العملاء والمنافسين والموردين والحكومة والتكنولوجيا والأوضاع الاقتصادية ... إلخ . أما البيئة الداخلية فتتضمن السيسات والأنشطة والتصرفات والقيود .. إلخ .

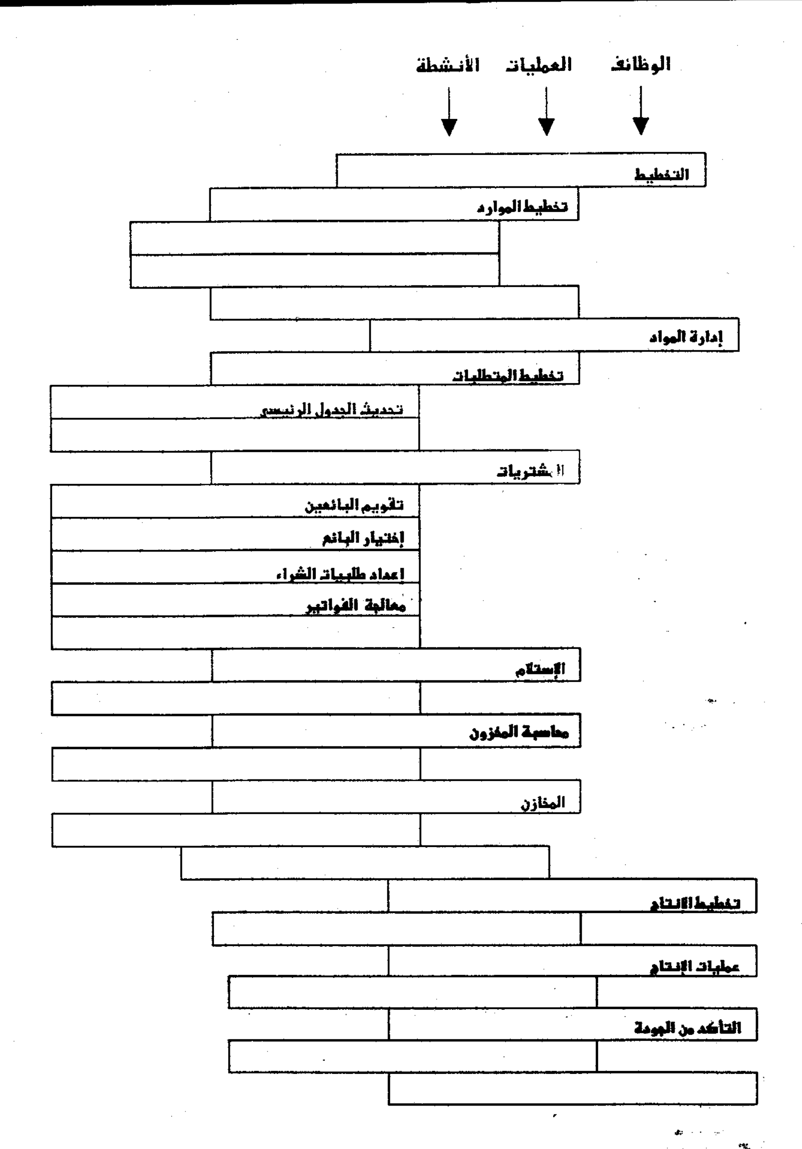
٢- تخطيط المنظمة :

يتضمن تخطيط المنظمة تطوير الأهداف والاستراتيجيات ومتطلبات الموارد والسياسات والقيود . وتعتبر خطة المنظمة التي يطلق عليها في بعض الأحيان الخطة الرئيسية وثيقة التخطيط الأساسية للمنظمة ، فهي تعد وتكتب بعناية فائقة وتوصل إلي كل المديرين الرئيسيين في النظمة . كما يجب أن تراجع وتحدث هذه الخطة الرئيسية عندما تتغير الأوضاع .

٣- تحليل نظم العمل :

يعتبر تحليل نظم العمل بالمنظمة الأساس الذي تبني عليه خطة قاعدة البيانات . والهدف من هذه العملية يتمثل في تحليل وظائف المنظمة الأساسية والفرعية على السواء ، وتعريف احتياجات المعلومات الحالية والمستقبلية التي تساند هذه الوظائف .

وتحليل وظائف المنظمة يعتبر عملية مستقلة عن خطوط التنظيم الرسمي بالمنظمة . ويستخدم في هذا التحليل المدخل الذي يستطرد من أعلي لأسفل أي من القمة إلي القاعدة التنظيمية . حيث يمكن تعريف الوظائف الرئيسية والتي يقسم كل منها إلي الوظائف الفرعية التي يطلق عليها العمليات ، وكل عملية من هذه العمليات قد تتفرع أيضا إلي وظائف فرعية أو ما يطلق عليه أنشطة . كما يعرف خلال التحليل كيانات المنظمة التي تحتاجها للقيام بكل عملية من العمليات المعرفة. ويمكن بيان معالم تحليل نظم العمل في شركة صناعية في الشكل التالي الذي قد يطلق عليه خريطة المنظمة :



شكل رقم (٦) خريطة تحليل نظم عمل شركة صناعية

وتمثل وظائف المنظمة مجموعات عريضة من الأنشطة والمهام والقرارات المترابطة التي تسهم في دورة حياة أي منتج أو خدمة . وكما اتضح من الشكل السابق فإن وظائف المنظمة تمثل التخطيط والمواد وتخطيط الإنتاج وعمليات الإنتاج والتأكد من الجودة Quality assurance التى تكون من الوظائف الأساسية فى كثير من الشركات الصناعية الكبيرة التى قد تمتد مجموعات وظائفها أكثر من عشرين

وظيفة . أما الشركات الصغيرة فتتراوح وظائفها من خمسة إلى عشرة مجموعات من الوظائف . وقد ترتبط إحدى وظائف المنظمة بوحدة تنظيمية قائمة بالفعل أو تتداخل مع وحدات تنظيمية عديدة قائمة . فعلى سبيل المثال قد تنتشر وظيفة " التأكد من الجودة" كما فى الشكل السابق خلال وحدات تنظيمية عديدة كالهندسة والمشتريات والرقابة على الجودة . على أنه يستحسن تصميم قاعدة البيانات بالإعتماد على الوظائف والعمليات والأنشطة بدلاً من الإعتماد على الوحدات التنظيمية ، إذ أن التفريغات التنظيمية تتغير على الدوام نتيجة لوجود كثير من المتغيرات المحيطة بالمنظمة والكائنى فى أعمالها .

أما العمليات فهى مجموعات الأنشطة التى ترتبط بالقرار وتحديث فى إطار الوظيفة ، وفى الغالب تخدم إدارة الأفراد أو الإدارة المالية أو إدارة المواد أو المعلومات . فكما فى الشكل السابق نجد أن وظيفة إدارة المواد تنقسم إلى عمليات مثل تخطيط المتطلبات والمشتريات هذا بالإضافة على عمليات الإستلام ومحاسبة المخزون أو المخازن . وبذلك فإن العمليات النابعة من الوظيفة تعكس تجميعات من الأنشطة المرتبطة بالوظائف الإدارية القائمة بالفعل . وكل وظيفة من ظائف المنظمة قد يعمل لها نموذج مستقل فى حد ذاته يشتمل من ثلاثة إلى عشر عمليات .

أما الأنشطة فى إطار العمليات فهى الأفعال أو التصرفات المعينة التى تتطلب لآداء عملية ما فمثلاً تتفرغ عملية المشتريات إلى أنشطة مثل تقويم البائعين وإختيار البائع أو الوكيل وإعداد طلبات الشراء ومعالجة الفواتير . والأفعال ما هى إلا إجراءات موجهة للأنشطة وتنتج بيانات تضاف إلى قاعدة البيانات أو تحديثها أو تحذف منها .

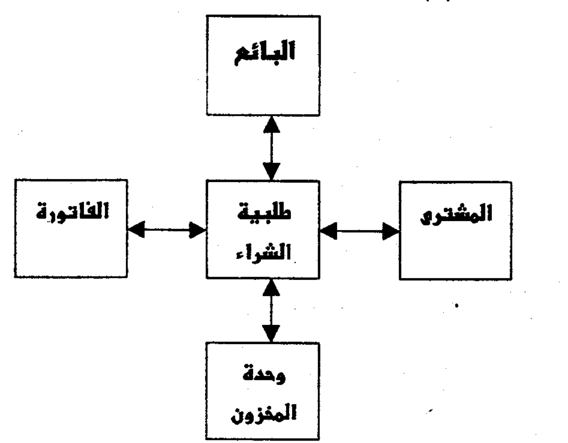
أما كيانات المنظمة Entities فتعبر عن الأفراد أو الأشياء أو الأحداث أى المعلومات التى تسجل فى قاعدة البيانات . وفى الإمكان تقرير الكيانات المطلوبة لإنشاء قاعدة البيانات عن طريق التعرف على كل العمليات المتضمنة فى الوظائف وتقرير الأنشطة والتساؤل عن الكيانات المطلوبة للإدخال والمعالجة والإخراج لكل نشاط . فعند تداول عملية المشتريات مع المستخدمين يمكن الكشف عن الكيانات المطلوبة لذلك والتى يمكن تحديدها مثل البائع وطلبية الشراء والفاتورة والمشترى ووحدة المخزون .

٤- تحليل قاعدة البيانات :

يعتبر تحليل قاعدة البيانانت المرحلة الأخيرة فى عملية تخطيط نظم المنظمة حيث تعرف الكيانات المختلفة المطلوبة لآداء الأنشطة والعمليات والوظائف التى سبق تعريفها فى العملية السابقة . وبذلك فإن هدف تحليل قاعدة البيانات يتمثل فى تعريف العلاقات بين الكيانات المؤثرة التى تساعد فى تطوير نموذج فكرى للمنظمة يشار إليه كنموذج للمنظمة ، ويطور أيضاً خلال هذه العملية أولويات عمل وجداول لقاعدة البيانات . كما تساند خطط قاعدة البيانات أولويات العمل والجداول الخاصة بنظم المعلومات المتكاملة والشمولية على مستوى المنظمة ككل .

والشكل التالى يوضح نموذج محدد لبيانات كيانات قاعدة البيانات الخاصة بالمشتريات .

شكل رقم (٧) نموذج بيانات كيانات قاعدة بيانات المشتريات :



النموذج السابق بيانه يشتمل على الكيانات الخاصة بالبائع والمشترى وطلبية الشراء ووحدة المخزون والفاتورة . وتتضمن هذه الكيانات وغيرها فى إطار عملية المشتريات ، ويمكن تحديد العلاقات بين هذه الكيانات كما يلى :

١- علاقة البائع بطلبية الشراء:

قد تشتمل هذه العلاقة على علاقة واحدة أو عدة علاقات حيث أن لكل بائع طلبيات شراء عديدة قد تتواجد فى وقت واحد . إلا أنه توجد علاقة واحدة لطلبية شراء معينة مع بائع واحد فقط .

٢- علاقة المشترى بطلبية الشراء:

كما فى العلاقة السابقة ، فقد تكون هذه العلاقة واحدة أو متعددة . وقد يكون للمشترى الواحد عدة طلبيات شراء . إلا أن طلبية الشراء الواحدة تكون لمشترى واحد فقط .

٣- علاقة طلبية الشراء بالفاتورة :

قد يكون هناك عدد من الفواتير لكل طلبية شراء تسلم بواسطة البائع ، إلا أن الفاتورة المسددة تطبق على طلبية شراء واحدة .

٤- علاقة طلبية الشراء بوحدة المخزون :

يمثل هذا نوعا من العلاقات المتعددة . فقد يكون لكل طلبية شراء وحدات طلبات عديدة ، كما قد يتواجد لكل وحدة مخزون طلبيات شراء متعددة تحت الطلب فى أى وقت معين .

حالة تخطيط قاعدة البيانات لإحدى المستشفيات

على الرغم من أن المثال الخاص بهذه الحالة هو مثال نظرى ، إلا أنه يشتمل على كثير من عناصر بيئة مستشفى واقعى إلى حد كبير . المستشفى فى هذه الحالة هو إحدى مستشفيات التأمين الصحى يشتمل على ما يقرب من ١٠٠ سرير أى يعتبر ذا حجم متوسط ويوجد فى عاصمة إحدى محافظات الوجه القبلى ويخدم أكثر من مائة ألف موظف وعامل يعملون فى هذه المحافظة ومؤمن عليهم صحياً

ومن الناحية التنظيمية يقسم المستشفى إلى مجموعتين أساسيتين كما هو الحال فى تنظيم معظم المستشفيات . فالمجموعة الطبية الكلينيكية تحت رئاسة مدير المستشفى وهو المدير الفنى فى نفس الوقت . والأطباء فى هذه المجموعة مسئولون مسئولية مباشرة عن جودة العملية الطبية والعناية الطبية المقدمة للمرضى . أما المجموعة الثانية من الأفراد العاملين فى التمريض والرعاية الطبية والإداريين والفنيين الذين أعمال الأطباء فى المجموعة الأولى فهم فى خدمة المرضى . وعلى الرغم من أنهم يتبعون نهائياً مدير المستشفى إلا أنهم تحت الرئاسة المباشرة للمدير الإدارى بالمستشفى الذى يتبع مباشرة مدير المستشفى .

تخطيط العمل :

للهيئة العامة للتأمين الصحى التى تتبعها هذه المستشفى خطة طويلة الأجل تمتد إلى عشرة أعوام من الآن تعرف مجال خدمات المستشفى والنمو المتنبأ به لها ، كما تحدد الأهداف الرئيسية للمستشفى ، وتعرف قدرتها على الإستيعاب والتوسع والمواد المتطلبة لها لكى تواجه متطلبات المستقبل .

وعلى الرغم من أن معظم المرضى المترددين على المستشفى يقطنون المدينة التى تتواجد بها إلا أنها تقبل المرضى من المدن الأخرى بالمحافظة وفى بعض الأحيان

من المحافظات الأخرى . وبذلك فإن مجال خدمة المستشفى هو المحافظة التى تتواجد بها والتى يربو سكانها على مليونين نسمة بنسبة نمو ٣% سنويا ومتوقع أن يستمر هذا النمو فى المستقبل . كما أن الموظفين والعاملين الذين تخدمهم المستشفى متوقع أن يزداد عددهم بنسبة ٥% فى المستقبل وتتوسع الخدمات لعائلاتهم أيضا .

وبذلك فإن الهدف الرئيسى لهذه المستشفى هو إستمرار مواجهة إحتياجات المحافظة التى تتواجد فيها من الرعاية الصحية للعاملين بها بجودة عالية بينما تبقى على معدل التكاليف المستمر فى الزيادة كما هو طبقا لخطط ضغط المصروفات وترشيد النفقات إلى حد كبير .

وحتى يمكن للمستشفى من مساندة الطلبات المتوقعة على خدماتها فإن خطتها الطويلة الأجل تتطلب التوسع فى التسهيلات المتاحة بالمستشفى وتحديثها . وتشتمل هذه الخطة على إضافة جناح جديد يتوسع عدد أسرة المستشفى بخمسين سرير فى الخمس سنوات التالية ، وبالفعل تتواجد أراضى مناسبة لهذا التوسع بالإضافة إلى أماكن إنتظار للسيارات والشاحنات . أما التسهيلات المتاحة حاليا فيجب أن تجدد وتحدث بحيث تشتمل على أماكن إستقبال وعيادات خارجية ملائمة . وقد خطط لإدخال تسهيلات جديدة للعلاج الطبيعى والأشعة المهبطية Ultrasoinc .

نظم المعلومات الحالية بالمستشفى :

تبنى نظم المعلومات الحالية على أساس نظم معالجة البيانات ذات الوجهة المعتمدة على الحزم Batch-oriented ، التى تشتمل على برامج تطبيق لمحاسبة المرضى وإعداد الفواتير والحسابات المستلمة والمحاسبة المالية . وتطبق حزم التطبيقات على معظم مستشفيات الهيئة العامة للتأمينات الصحية ، إلا أن أجهزة الكمبيوتر لا تتوافر فى معظمها فى حين يمكن القيام بالتطبيقات فى مراكز خدمةمستقلة كأجهزة الكمبيوتر التى قد تتواجد فى ديوان عام المحافظة أو الجامعة بها أو أى هيئة تقدم هذه الخدمات . وفى تخطيط الهيئة الطويل الأجل توحيد هذه التطبيقات وإدخال أجهزة الكمبيوتر الصغيرة فى هذه المستشفيات وربطها أيضا بشبكة كمبيوتر متطورة فى هذا الصدد .

وعلى الرغم من أن المستشفى الذى نقوم بدراسته يتوفر على جهاز كمبيوتر متوسط الحجم ومزود بحزم البرامج التطبيقية المشار إليها إلا أنه يتسم بالقصور والجمود لما يلى :

- عدم مساندة نظم الأطباء من حيث تسجيل وإعداد تقارير مخرجة عن نتائج الإختبارات والإجراءات المعملية المختلفة .
- لا تساند برامج ذات الوجهة المعتمدة على الحزم إجراءات الوصول المباشر الخاصة بتسجيل المرضى وإجابة الإستفسارات المتصلة بالفواتير على سبيل المثال .
- عدم تصنيف التكاليف على أساس مركز التكلفة أو الوحدات التطبيقية المختصة .
- عدم مرونة النظام لبقية الإحتياجات المتغيرة أو المتغيرات المتكررة لإعداد التقارير للهيئات الصحية الخارجية .

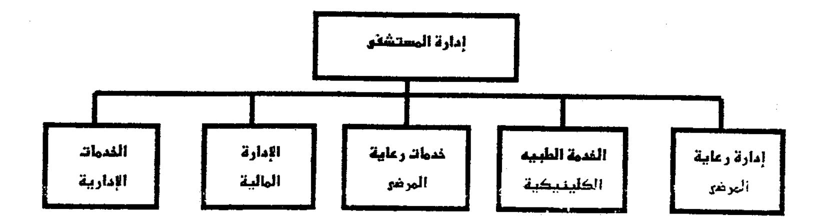
وقد إكتشفت إدارة المستشفى أن نظام المعلومات الحالى المطبق فى المستشفى لا يستجيب لإحتياجاتها المتنوعة والمتغيرة بصفة مستمرة .. وقد بدأت الإدارة بإدخال أسلوب نظم إدارة قاعدة البيانات وتعيين " مدراى قاعدة البيانات " . ولكن الإدارة طلبت من العاملين فى قسم نظم المعلومات إعداد دراسة جدوى لإدخال هذا الأسلوب لتحليل التكاليف والعوائد وتطوير خطة شمولية لقاعدة البيانات . وبالفعل شكل فريق عمل أو فريق دراسة لإعداد ذلك من العاملين بالمستشفى بمساعدة مستشار خارجى .

وعلى مدى أسبوع عمل متواصل أعد فريق العمل إطار الدراسة المقترحة والجداول الخاصة بها التى تشتمل على :

تحليل نظم العمل بالمستشفى :

راجع الفريق الخطة الطويلة الأجل للمستشفى وتحليل نظم العمل بها . وقد قام فى هذا الصدد بتعريف الوظائف الرئيسية للمستشفى التى يمكن توضيحها فى الشكل التالى :

شكل رقم(٨) الوظائف الرئيسية فى المستشفى



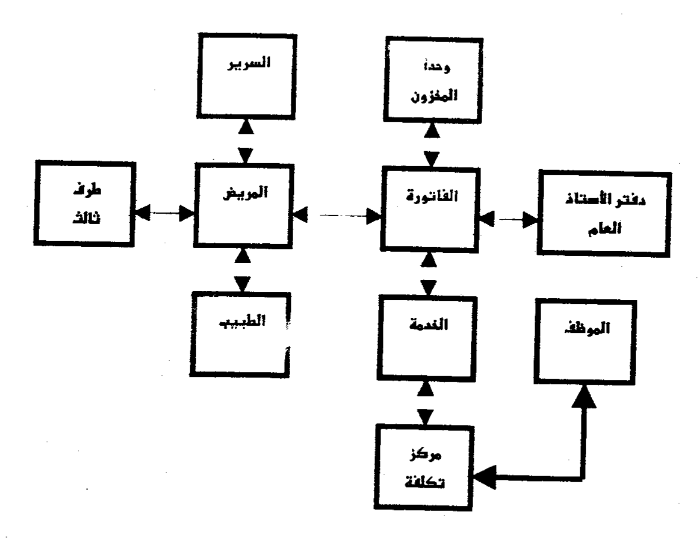
- إدارة رعاية المرضى الخاصة بجدولة المرضى وتسجيلهم وإجراءات الكشف عليهم وتشخيص الأمراض والدخول والخروج أو التحويل .. إلخ .
- الخدمة الطبية الكليينيكية التى تتعلق بالإختبارات والتحليلات والأشعات والمراجعة .
- خدمات رعاية المرضى الخاصة بالتمريض والعمليات الجراحية والعلاج الطبيعى .. إلخ .
- الإدارة المالية وتتصل بمحاسبة المرضى وإعداد التكاليف وكشوف المرتبات والأجور والمحاسبة العامة .
- الخدمات الإدارية التى تساند إدارة المستشفى من المشتريات والرقابة على المخزون وإدارة الأفراد ... إلخ .

بعد التعرف على الوظائف الأساسية تمثلت الخطوة التى تلت ذلك تفسير العمليات لكل وظيفة . قد أعدت خريطة توضح نموذج وظائف والعمليات وبعض الأنشطة كما هو فى الشكل التالى :

شكل رقم (٩) خريطة نموذج العمل بالمستشفى

وبواسطة هذه الخريطة ، تمكن فريق العمل من تعريف الوظائف والعمليات والأنشطة المختلفة التى تؤدى فى المستشفى . وبعدئذ قام الفريق بتحديد الكيانات الأساسية المطلوبة للعمليات كما هو موضح فى الشكل التالى :

شكل رقم (١٠) نموذج الكيانات الأساسية



وهذه الكيانات العشرة التى أمكن تعريفها وتحديدها فى هذه المرحلة تمثل موارد البيانات الأساسية المطلوبة للمستشفى .

وقد أكمل فريق الدراسة تحليل النظم المتواجدة فى المستشفى وإعداد تقرير مختصر عنها عرض لإدارة المستشفى مشتملا على النتائج المتوصل إليها. هذا المدخل المتكامل فى تخطيط نظم المعلومات كأن ذا تأثير على الإدارة العليا بالمستشفى والهيئة العامة للتأمين الصحى التى أوصت الأخذ به وإستمرار الفريق فى تطوير خطة قاعدة البيانات بالمستشفى .

يتضح من الشكل السابق الذى توصل إليه فريق العمل المخصص لهذ الدراسة التخطيطية والذى يتضمن نموذج العمل بالمستشفى أنه يشتمل على عديد من العلاقات والإرتباطات التى يمكن توضيحها فيما يلى :

١- علاقة المريض بالسرير :

وهى علاقة ذا صفة فردية فى وقت معين حيث يخصص لكل مريض سرير واحد والعكس صحيح .

٢- علاقة المريض بالفاتورة :

وهذا الإرتباطية ذا صفة فردية أيضا حيث أن الفاتورة الواحدة ترتبط بمريض واحد فقط .

٣- علاقة وحدة التكلفة بمركز الخدمة :

تمثل هذه العلاقة إرتباط شئ واحد أو خدمة واحدة بأشياء عديدة . فلكل مركز تكلفة كالأشعة على سبيل المثال تتوفر وحدات خدمية عدية ، ولكن فى نفس الوقت ترتبط وحدة الخدمة المعينة بمركز تكلفة واحد فقط .

٤- علاقة دفتر الأستاذ العام General Ledger بفاتورة المريض :

ويمثل ذلك علاقة مريض واحد بعديد من الإرتباطات ، فدفتر الأستاذ العام يشتمل على عديد من الحسابات المختلفة . ولكن كل فاتورة مريض ترتبط بحساب واحد مثل الحسابات المستلمة .

٥- علاقة الطبيب بالمريض :

يمثل ذلك إرتباط ذا علاقات متعددة فلكل طبيب عدد معين من المرضى كما أن لكل مريض معين أكثر من طبيب معالج له .

٦- علاقة الطرف الثالث Third Party بالمريض :

وهى علاقة واحد بكثير من الإرتباطات . ولكل طرف ثالث كبنك الدم أو الصيدلية أو المنظمة المؤمن لها مجلسين كثيرين ، ولكن يفترض لكل مريض أن يكون له طرف ثالث معين .

٧- علاقة الفاتورة بوحدة الخدمة :

يمثل ذلك إرتباط ذات علاقات متعددة لكـل مـن طرفـى العلاقـة فتشـمل الفـاتورة علي وحدات خدمة عديدة بينما تتواجد وحـدة الخدمـة فـى فواتـير عديـدة للمرضـى فـى وقت معيـن .

وبذلك فعن طريق تخطيط هـذا النمـوذج كمـا بيـن فـى الشـكل السـابق شـرحة وتحديد علاقاته فإن ذلك يؤثر عى تحليل متعمـق يتصـل بـالإجراءات وقواعـد الأداء التـى تتحكم فى تحديد الإرتباطات بين الكيانات الخاصـة بالعمليـات فـى تخطيـط تصميـم قواعـد البيانات بالمنظمـة .

التقرير النهائى :

وقد أعد فريق الدراسـة تقريـرا نـهائيا لخـص فيـه قـاعدة البيانـات للمستشـفى وإشتمل هذا التقرير على خريطة عمل لوصـف الوظـائف والعمليـات والأنشـطة والكيانـات المتضمنة فـى المستشـفى ، هـذا بالإضافـة إلـى نمـوذج المستشـفى ، وقائمـة بتحليـل التكاليف والعوائد التى تعود على المستشفى من إدخـال أسـلوب قـاعدة البيانـات ، وخطـة العمل المطلوب آدائها فى تصميم وتنفيذ قـاعدة البيانـات المطلوبـة للمستشـفى .

وكان القرار النـهائى لإدارة المستشـفى هـو إعتمـاد الخطـة المقدمـة والوافقـة علـى تعييـن أحـد خـبراء المعلومـات كـإدارى قـاعدة البيانـات Database Administrator .

أخير يمكن القول فى نهاية هـذا الفصـل أن مرحلـة تخطيـط قـاعدة البيانـات فـى أى منظمة تعد مرحلة أساسـية لتصميـم قـاعدة البيانـات ، حيـث يعتـبر تخطيـط قـاعدة البيانات من الأساليب الجوهرية التـى توليـها المنظمـة جانبـا كبـيرا مـن الأهميـة لجنـى ثمار ومزايا إستخدام وإدارة موارد البيانات بها . وبذلـك فـإن خطـة قـاعدة البيانـات تعـد جزءا هاما وحيويا من الخطة العامة لنظام معلومـات المنظمـة المتكـامل .

الفصل الثامن

مفاهيم متقدمة
فى بيئة قواعد البيانات

الفصل الثامن
مفاهيم متقدمة فى بيئة قواعد البيانات

فى هذا الفصل سوف نتعرض لبعض المفاهيم المتقدمة والهامة فى مجال إستخدام المفهوم الخاص بنظم قواعد البيانات والتى يمكن تناولها على النحو التالى :

(١) خدمات قواعد البيانات التجارية COMMERCIAL DATABASE SERVICES

تبيع خدمة قواعد البيانات (التجارية) فى الخط المفتوح (Online commercial database) الإتصال بقواعد بيانات كبيرة (عادة على المستوى القومى) . ويمكن أن تضيف مثل هذه الخدمة بيانات خارجية إلى MSS بطريقة موقوتة وبتكلفة معقولة . كل ماهو مطلوب لإسترجاع البيانات من مثل هذه الخدمة هو نهاية طرفية ، ومود ، وهاتف ، وكلمة مرور ، وبعض رسوم للخدمة . وقد اصبحت هذه الصغية من مورد المعلومات ، التى تسمى فى بعض الأحيان بنك معلومات محوسب ، سعبية للغاية . ويتاح حالياً عدة آلاف من هذه الخامات [1]. ويسرد جدول (١) العديد من هذه الخدمات . كما تقدم بعض هذه الخدمات إمكانيات مشاركة زمنية لإستخدام منتجات DSS .

---

(١) يعد Directory of Online Databases دليلاً يصدر كل ربع سنوية من قبل Cuadra Assoc .,Inc (Santa Monica, CA (with Elsevier Publishing Co., New York) والذى يوفر معلومات حديثة عن قواعد البيانات التجارية . كما يتاح هذا الدليل فى الخط المفتوح أيضاً .

١/١ الإتصال الذكى بقواعد بيانات فى الخط المفتوح

Intelligent Access To Online Databases.

طورت قواعد البيانات التجارية فى الخط المفتوح مستقلة عن بعضها بعضاً ، ولها لغات أوامر ، وهياكل ملفات وبروتوكولات إتصال مختلفة (عدة آلاف) ، وإفتقار للنمطية، وليس عجباً رؤية لمذا توجد حاجة لمعرفة موسعة ليمكن إستخدام قواعد البيانات هذه بكفاءة . ويمكن إستخدام نظم الخبرة (عادة مدمجة مع مشغل لغة طبيعية) كسطح بينى لمثل قواعد البيانات هذه . وتشمل قاعدة العرفة لنظم ES معرفة عن إستراتيجية للبحث . مثال ذلك ، يمكن أن ينصح مثل هذا النظام مستخدم لانظامى عن كيفية إجراء بحث بسيط ، أو يمكن أن يوجه المستخدم الأكثر خبرة فى الإتصال بقواعد بيانات لها تنظيمات معقدة . وفى الكل ، يمكن أن يجعل ES نظام الخط المفتوح مفتولاً إلى المستخدم . ويكون مثل هذا التكامل فى غاية الأهمية لنظم المعلومات التى يتكرر إستخدامها كقواعد بيانات خارجية .

جدول (١) خدمات قواعد بيانات تجارية .

CompuServe and The Source. Personal computer networks providing statistical data banks (business and financial market statistics; as well as bibliographic data banks (news, reference, library, and electronic encyclopedias). CompuServe is the largest supplier of such services to personal computer users.

Compustat. Provides financial statistics about more than 12,000 corporations.

Data Resources, Inc. Offers statistical data banks in agriculture, banking, commodities, demographics, ecomonics, energy, finance, insurance, international business, and the steel and transportation industries. DRI economists maintain a number of these data banks. Standard & Poor's is also a source. It offers services under the U.S. Central Data Bank.

Dow Jones Information Service. Provides statistical data banks on stock market and other financial markets and activities, and in-depth financial statistics on all corporations listed on the New York and American stock exchanges, plus 800 other selected companies. Its Dow Jones News/Retrieval system provides bibliographic data banks on business, financial, and general news from *The Wall Street Journal*, *Barron's*, the Dow Jones News Service, *Wall Street Week*, and the 21-volume *American Academic Encyclopedia*.

Interactive Data Corporation. A statistical data bank distributor covering agriculture, autos, banking, commodities, demographics, economics, energy, finance, international business, and insurance. Its main suppliers are Chase Econometric Associates, Standard & Poor's, and Value Line.

Lockheed Information Systems. The largest bibliographic distributor. Its DIALOG system offers extracts and summaries of more than 150 different data banks in agriculture, business, economics, education, energy, engineering, environment, foundations, general new publications, government, international business, patents, pharmaceuticals, science, and social sciences. It relies on many economic research firms, trade associations, and governmental groups for data.

Mead Data Central. This data bank service offers two major bibliographic data banks. Lexis provides legal research information and legal articles. Nexis provides a full-text (not abstract) bibliographic database of over 100 newspapers, magazines, and newsletters, news services, government documents, and so on. It includes full text and abstracts from the *New York Times* and the complete 29-volume *Encyclopedia Britannica*. Also provided is the Advertising & Marketing Intelligence (AMI) data bank, and the National Automated Accounting Research System.

المصـــدر: Stephen L. Alter, *Decision Support Systems*, 1980 by Addison-Wesley Publishing Company, Inc.

٢/١ قواعد البيانات الشيئية OBJECT - ORIENTD DATABASES

تحتاج تطبيقات MSS المعقدة مثل تلــك التــي يتضمنــها نظــام التصنيــع المتكــامل بالحاسـب (CIM) Computer Integrated manufacturing إلـــى إمكانيـــة الإتصـــال ببيانات مركبة ، والتى يمكـــن أن تشـــمل صـــورا ، وعلاقــات معقــدة . ولايمكـــن أن تكـــون معمارية قواعد البيانات العلاقية . أو الهرميـــة ، أو الشـــبكية كفـــؤة لتنـــاول قواعـــد بيانـــات معقدة . حتى عندما تســـتخدم SQL فـــى إنتـــاج قواعـــد بيانـــات علاقيـــة والإتصـــال بـــها ، يمكن ألا يكون الحل فعالاً . فكـــل قواعـــد البيانـــات ســـالفة الذكـــر قواعـــد بيانـــات حرفيـــة عـــددية . وفى بعض الأحيان تأخذ تمثيلاً رسومياً لتحقيـــق أفضـــل نتـــائج .

وتعتمد إدارة البيانات الشيئية على مبـــدأ البرمجـــة الشـــيئية . وتدمـــج نظـــم قواعـــد البيانات الشيئية خواص لغة برمجة شـــيئية مثـــل Smalltalk ، أو C++ مـــع آليـــة لتخزيـــن البيانـــات والإتصـــال بـــها . وتركـــز الأدوات الشـــيئية علـــى قواعـــد البيانـــات مباشـــرة . وتسمح قـــاعدة البيانـــات الشـــيئية Oriented database بتحليـــل البيانـــات علـــى مســـتوى مفاهيمى يركز علـــى العلاقـــات الطبيعيـــة بيـــن الأشـــياء . ويســـتخدم التجريـــد فـــى عمـــل هرميات إرث ، ويسمح تغليف الشئ لمصمـــم قـــاعدة البيانـــات بتخزيـــن كـــلا مـــن البيانـــات التقليدية ، والشفرة الإجرائية فـــى نفـــس الأشـــياء . ويتوفـــر لـــدى نظـــم قواعـــد البيانـــات الشـــيئية القوة والقـــدرة علـــى تنـــاول البيانـــات المعقـــدة المســـتخدمة فـــى تطبيقـــات MSS المعقـــدة.

ويعرف نظام إدارة البيانـــات الشـــيئى : البيانـــات كأشـــياء ، وبيانـــات مغلفـــة مـــع هيكلها وسلوكها المناســـبين . ويســـتخدم النظـــام هرميـــة مـــن الطبقـــات ، والطبقـــات مـــن الأشياء . ويكون الـــهيكل ، بالنســـبة إلـــى العلاقـــات ، والســـلوك ، بالنســـبة إلـــى الطـــرق والإجراءات موجودين كمحتويـــات للشـــئ .

٣/١ دعم قرار المنشأة ومستودع البيانات :

ENTERPRISE DECISION SUPPORT AND THE INFORMATION WAREHOUSE

تكون القوى المتعارضة واضحة فى الأعمال الحديثة (طبقاً لعالم الحاسب Computerworld May 17,1993,page53) وتتمثل أهم تلك القوى فى أنها : أولاً ، هناك حاجة لحلول نظم برامج ونظم مكونات متخصصة ومحلية . ثانياً ، هناك حاجة لوسيلة فعالة من ناحية التكلفة لتوحيد Uniting موارد المعلومات هذه فى أصول الأعمال التى يمكن إدراتها . ومع إستجابة المديرين لهذا التحدى ، يجب عليهم أن يعملوا مع نمو متزايد فى عدد وتنوع الوحدات والنظم . وتصبح نظم المنشآت أكثر تعقيداً جداً . ولدى المنظمات فى وقتنا الحالى خليط من النظم المركزية القديمة ، والنظم المنتشرة الحديثة - مدى واسع من التقنيات المختلفة يقدمها عدد كبير من المنتجين - ويقع التحدى فى إدارة هذا الخليط من الموارد . وأحد طرق التعامل مع مثل هذه الإدارة هو مفهوم حوسبة المنشأة .

يشار إلى نظم الحوسبة التى تشمل المنظمة كلها بأنها حوسبة المنشأة Enterprise Computing ، أو نظم على مستوى المنشأة Enterprise - Wide Systems .(يشير مصطلح منشأة Enterprise إلى المنظمة كلها).

وتعد حوسبة المنشأة معمارية لنظام حاسب متكامل يخدم إحتياجات الأعمال للمنشأة . وهو إطار تقنى يحتوى تطبيقات متعددة Muliple Applications ، ونظم مكونات ، ونظم برامج ، وقواعد بيانات ، وشبكات ، وأدوات إدارة ، من نظم التشغيل وحتى بروتوكولات الإتصالات . كما أنه يتطلب إجماع أيضاً على رصيف إدارة مشتركة مفتوح ، وتعهد تنظيمى قوى . وفيما يلى الفوائد الرئيسية لحوسبة المنشأة :

- مستويات مرتفعة مستمرة من الخدمة العولية والمستجيبة .
- سهولة إدخال حلول الخادم والعميل مع مناهج الحاسب الكبير الموجودة لذلك ، يمكن إستخدام التشغيل التعاونى وتعزيزه للتطبيقات الموجودة .

- يمكن إدخال تغييرات متكررة وسـريعة ، وزيـادة في التعقيـد بسـرعة ، دون أن يتعرض النظام ، أو الشـبكة لأى خطـر .
- تضمن الأمثلية الأكبر للشـبكة ، ومـوارد النظـام الإحتفـاظ بالخدمـة مرتفعـة الجودة عند أقل مستوى تكـاليف ممكـن .
- تسمح تلقائية العمليات الإدارية بإنخفاض تسـيير النظـام ، وتكـاليف التشـغيل بإستمرار مع زيادة نمو المنشـأة .
- تعزز الشبكات وامن البيانـات .

مفهوم مستودع المعلومات The Concept of Information Warehouse :

مفتاح الإستخدام الناجح لنظـم MSS علـى مسـتوى المنشـأة كلـها هـو البنيـة التحتيـة التـى تدعـم الإتصـال ، والإسـترجاع ، والمعالجـة ، والتحليـل ، وإعـداد العروض (التقديمات) ،ومنهجيات عروض الرسـومات ، وتوصيـل (نقـل) نتـائج هـذه السلسلة من الأنشطة . وتشمل الموضوعات الأكـثر أهميـة إتاحـة البيانـات الدقيقـة ، والمتسقه ، والموقوتة ، والقابلة للاستخدام . لقـد كـان المديـرون ، وعمـال المعرفـة الآخرون الذين يعدون مستخدمين أوليين لنظـم MSS المسـتخدمة ، أو لعـدم وجـود دليل لإتاحة ، وحداثة البيانات اللآزمة للاسـتخدام . وبـإدراك هـذا الموقـف ، أنتجـت شـركة IBM مفـهوم مسـتودع البيانـات Information Warehouse(IW) ، والـذى صمم للسماح للمنشآت " بعدم منع " البيانـات التـى كـانت ، حتـى وقـت قريـب غـير متاحة لمسـتخدمى MSS.

وتعرف IBM مستودع المعلومات بأنـه مجموعـة مـن DBMS وأسـطح بينيـة ، وأدوات ، وتسهيلات تديـر وتتسـلم معلومـات موقوتـة ودقيقـة ومفهومـة لاتخـاذ قرارات الأعمال . ويعد IW إطارا يـهدف الـى توفـير درجـة إتصـال DSS مرتفعـة جدا بالبيانات عما كان ممكنـا مـن قبـل فـى معظـم تشـييدات نظـم المعلومـات IS ويحتوى IW على إستراتيجية عريضة ، ونمطيـات ، وأسـطح بينيـة ومنتجـات نظـم برامج من كل من IBM وموردين مسـتقلين مختـارين .

ويحتوى منـــهج IW علـــى ٣ مســـتويات : مســـتوى بيانـــات المنشـــأة enterprise data level يشمل DBMS الذى يحتوى على بيانـــات المنظمـــة . ويحتـــوى علـــى كـــل مـــن بيانات تشغيلية أو إنتاج تعتمـــد علـــيها العمليـــات اليوميـــة للأعمـــال ، وبيانـــات معلوماتيـــة والتى استخلصت لأغراض القيمة الدضافة لذلك فـــهى مفيـــدة فـــى اتخـــاذ القـــرار لكـــل مـــن منفذى الادارة العليا والعـــاملين المـــهنيين . وتشـــمل منتجـــات IBM التـــى تدعـــم مســـتوى بيانات المنشأة نظـــام إدارة قواعـــد بياناتـــها العلاقـــى المتفـــق عليـــه (DB2 ، SQL/DS و OS/2 Database manager) و DBMS الهرميـــة للبيانـــات التشـــــــغيلية (IMS ، و DL/I).

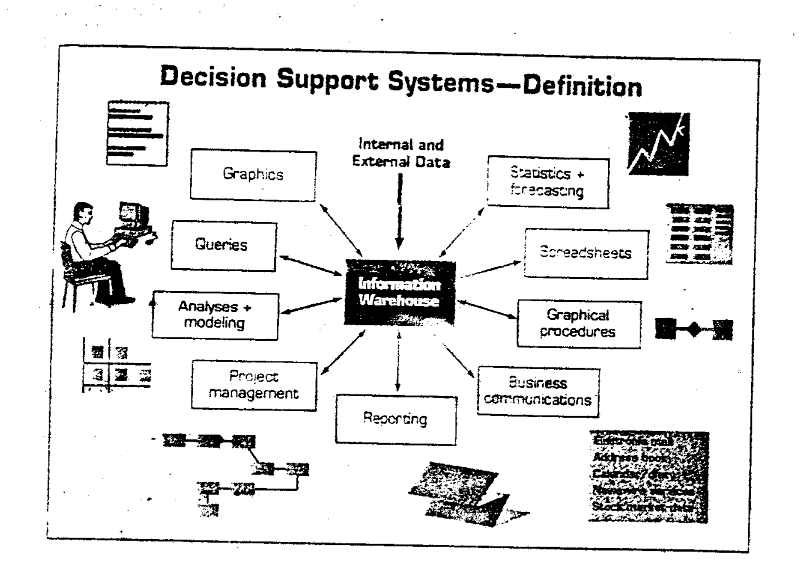
ويشـــمل مســـتوى دليـــل البيانـــات data directory level قطعـــة اتصـــال ونقـــل البيانات من IW . ويتكون هذا العنصر من منتجـــات مـــن كـــل مـــن IBM ، وشـــركائها فـــى الأعمال . وتقدم IBM ثلاثة منـــاهج مختلفـــة لتســـليم البيانـــات الـــى IW : اتصـــال بيانـــات بعيد Remote Data Access(RDA) والذى يدعمـــه تنفيـــذ IBM لاتصـــال البيانـــات العلاقـــى المنتشـــر Relation Data Access(DRDA) لنظمـــها DBMS العلاقيـــة الأربعـــة ، وكذلـــك اتصـــــال بيانـــــات المنشــــأة /لغــــة الاستفســـــار المهيكلـــــة Enterprise Data Access/SQL(EDA/SQL)، وهـــى عائلـــة مـــن منتجـــــات شـــــركة بنـــــاة المعلومـــــات Information Builders Inc.

ويشمل مستوى دعم القـرار Decision Support Level مـــن IW عـــددا كبـــير مـــن منتجـــات IBM . بإســـتثناء الســـطح البينـــى للمســـتخدم النـــهائى المعـــروف بإســـم مديـــر الاستفسار Query Manager الـــذى قـــد لـــ SQL/400 لحاسبـــات AS/400 ، توجـــد هـــذه المنتجات فى السوق بـــالفعل .

وجدير بالذكر أنه يمكـــن لمســـتودع المعلومـــات أن يتصـــل بالبيانـــات فـــى أنـــواع متعددة من البيانات المحليـــة أو المنتشـــرة غـــير المتجانســـة (لـــها أشـــكال مختلفـــة) علـــى مستوى المنشأة كلها عـــبر إدارة بيانـــات متعـــددة للمنشـــأة ، وتســـليم البيانـــات ، ونمذجـــة البيانات ، وتحليل البيانات ، وعناصر نظـــم برامـــج دعـــم القـــرار لمســـتودع المعلومـــات . ويتمركز مستودع البيانـــات حـــول معماريـــات قواعـــد البيانـــات العلاقيـــة عليـــه (DB2 ، SQL/DS و OS/2 Database manager وأكـــثر مـــن ٣٠ نـــوع آخـــر مـــن تخزيـــن /وإدارة

قواعد البيانات غير IBM) . وتكمن القيمة الحقيقية لمستودع المعلومات فى أنه يوفر اتصالا بالبيانات ، كما يحفظ أيضا فى نفس الوقت الأمن / والحماية / والاتساق فى نظم إدارة البيانات التشغيلية التى تكون حيوية لعمليات أعمال المنشأة اليومية . ومع تقديم معماريات قواعد البيانات المنتشرة (الموزعة) سيصبح MSS طريقة للحياة فى معظم المنظمات التى تنفذ إطارا مثل مستودع المعلومات .

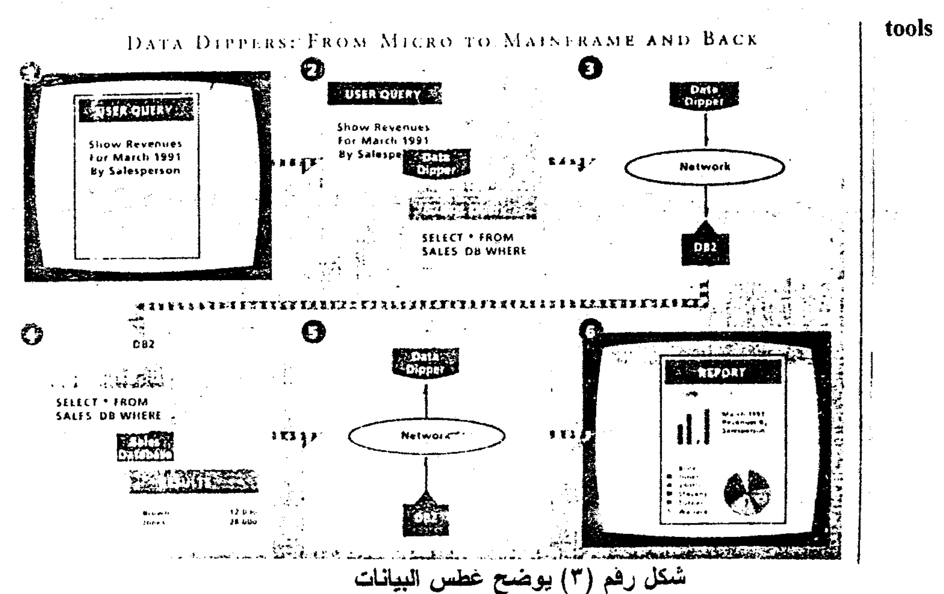
ولقد أدرك العديد من الموردين أهمية مثل هذا التكامل ، وطوروا منتجات نظم برامج لدعمه . ويأتى أحد أمثلة هذا المنتج Oracle relational DBMS من شركة Oracle Corp. ، الذى يدخل بعض وظائف ومنافع إستخدام ES فى صورة مثالية للاستفسار Query Optimizer الذى يختار المسار الأكثر كفاءة الذى تسلكه استفسارات قاعدة البيانات .



شكل رقم (٢) يوضح مستودع معلومات IBM

١/٤ الإتصال بالبيانات : غطس البيانات ACCESSING DATA : DATA DIPPING

ركزت تقنية المعلومات ، للعديد من السنوات على بناء نظم تدعم تشغيل العمليات الجارية للمنشأة . وتحتاج مثل هذه النظم أن تكون متسامحة فى الخطأ ، وتوفر إستجابة سريعة لمجموعة المستخدمين المتصلين بالقاعدة . حيث تركز تلك النظم على إنتاج العديد من الأدوات متعددة الأبعاد لتشغيل العمليات الجارية فى الخط المفتوح Online transaction processing(OLTP) وتعمل على دعم واتصال قرار الواجهة الأمامية ، وقاعدة بيانات الواجهة الأمامية ، ونظم إتصال المعلومات المرنية ، والتى يسميها البعض غاطسات البيانات data dippers لأنها تستخدم الغطس فى قاعدة البيانات لأخذ ماتكون هناك حاجة له .وتهدف هذه الأدوات الى مد المستخدمين بالقوة . لذلك ، تهتم التطبيقات التى لها هذه الطبيعة أكثر بكيفية إسترجاع ورؤية البيانات عن كيفية الحصول على البيانات ، وتخزينها . وتظهر عملية غطس البيانات data dipping فى الشكل رقم (٣) . وتظهر تحت أسماء مختلفة : نظم ذكاء الأعمال Business Intelligence Systems ، واتصال المستخدم النهائى بالبيانات end-user data access ، وأدوات الاتصال بالبيانات وعمل التقارير data access and reporting tools



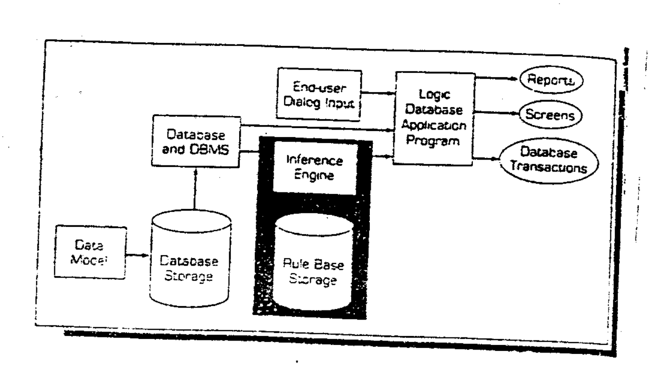
شكل رقم (٣) يوضح غطس البيانات

٥/١ قواعـد البيانـات الذكيـة Intelligent Databases

لقد أصبح تحديد المعلومات المرتبطـة بالمنظمـات الخاصـة والعامـة ، وإسـترجاعها ، وإستخدامها ، وحذفها أكثر تعقيدا مع زيـادة الكميـات . وفـى نفـس الوقـت ، يـزداد عدد الأفراد الذين يتداخلون مع هـذه المعلومـات بسـبب اسـتخدام الشـبكات ، وحوسـبة الخادم والعميل ، وانخفاض تكـاليف تشـغيل المعلومـات ، وأصبـح العمـل مـع قواعـد بيانات كبيرة مهمة صعبة تحتاج الى خـبرة كبـيرة .

ولقد ظهرت الحاجة الى نظـم قواعـد البيانـات الذكيـة كإطـار تكـاملى يجمـع بيـن قواعد البيانـات ، وانظـام الذكـاء الاصطنـاعى Artificial Intelligence(AI)(وخاصـة النظم الخبيرة) كأساس لتطوير وخدمة تطبيقـات نظـم دعـم الادارة MSS . حيـث يمكـن أن تجعـل تقنيـة AI ، خاصـة نظـم الخـبرة Expert system(ES) إسـتخدام وإدارة قواعد البيانات أبسـط .وإحـدى طـرق عمـل ذلـك هـى تعزيـز enhance نظـام إدارة قاعدة البيانات عن طريق توفير مقدرة سـطح بينـي لـه . ويسـمى هـذا النـهج قـاعدة بيانات ذكيـة intelligent databse .

ويوضح الشكل رقم (٤) تكامل كل من نظـم الخـبرة ES وقـاعدة البيانـات DB مـن أجل استخلاص برنامج التطبيق اللازم لخدمـة انظمـة دعـم الادارة MSS .



شكل رقم (٤)

يوضح قاعدة بيانات ذكية ، تبين طريقا واحدا لتكامل ES وقاعدة البيانات

الفصل التاسع

تطبيقات متقدمة باستخدام قاعدة البيانات ACCESS

الفصل التاسع

تطبيقات متقدمة باستخدام قاعدة البيانات ACCESS

مقدمة :

يعد برنامج Access ضمن سلسلة البرامج الجاهزة التى تخدم فى مجال التعامل مع قواعد البيانات والمفاهيم المرتبطة بها . ولقد ظهر هذا البرنامج كأحد حلقات سلسلة نظم ميكروسوفت لأتمتة المكاتب Ms-Office . ولقد خرج من هذا البرنامج عدة إصدارات آخرها هو الاصدار Access 2000 . وجميعها تخدم مجال قواعد البيانات سواء فيما يتعلق بإنشاء جداول أو نماذج أو الاستعلام ، أو التقارير ... الخ . وسوف نتناول شرح هذا البرنامج بشيئ من التفصيل على النحو التالى .

(١) بعض المفاهيم الأساسية فى مجال إستخدام برنامج Access :

**** البيانات : DATA**

هى مجموعة الحقائق التى تأخذ شكل أرقام أو حروف أو رموز أو كلمات قابلة للمعالجة بواسطة الحاسب مثل : الرقم (٦٥) أو كلمة بيانات .

**** المعلومات INFORMTION**

هى بيانات تم تنظيمها أو معالجتها لتحقيق أقصى إستفادة منها.

مثال : الرقم (٥) إذا إستخدما فى عملية الضرب ٥ x ٦ أصبحا معلومه مفيدة .

**** قواعد البيانات DATA BASE**

هى تجمع لكمية كبيرة من المعلومات أو البيانات وعرضها بطريقة أو بأكثر من طريقة تسهل الإستفادة منها .

مثال : دليل الهاتف الذى يشتمل على أسماء وعناوين وأرقام هواتف سكان مدينة القاهرة يمكن أن نعتبره قاعدة بيانات وتحقيق الإستفاده من قاعدة البيانات هذه بإدخال رقم المشترك والحصول على إسمه وعنوانه أو إدخال إسم المشترك والحصول على رقم هاتفه وعنوانه وهكذا .

**** نظم إدارة قواعد البيانات DATABASE MANAGEMENT SYSTEMS**

هى مجموعه من البرامج الجاهزة التى تقوم بتنفيذ جميع الوظائف المطلوبه من قاعدة البيانات .

مثال : بعد إضافة عملاء جدد لدليل الهاتف فى مدينة القاهرة فإنك قد تحتاج لإعادة ترتيب أسماء المشتركين أبجدياً أو لترتيب عناوينهم ، مثل هذا العمل من أحد وظائف إدارة قواعد البيانات .

(٢) أهمية قواعد البيانات :

أ. تخزين جميع البيانات بكافة الأنشطة لجهة ما بطرق متكاملة ودقيقة وتصنيف وتنظيم هذه البيانات بحيث يسهل إسترجاعها فى المستقبل

ب. متابعة التغيرات التى تحدث فى البيانات المخزنه وإدخال التعديلات اللآزمة عليها ، حتى تكون دائماً فى الصورة الملائمة لإستخدامها فور طلبها

ت. تخزين كم هائل من البيانات التى تتجاوز الإمكانيات البشرية فى تذكر تفاصيلها ومن ثم إجراء بعض العمليات والمعالجات التى يستحيل تنفيذها يدوياً .

ث. تساعد على تخزين البيانات بطريقة متكاملة ، بمعنى الربط بين النوعيات المختلفة للبيانات المعبرة عن كافة الأنشطة .

ج. تساعد على تحقيق الرية الكاملة للبيانات المخزنة بها بحيث لاتتاح أية معلومات لأى شخص ليس له الحق فى الإطلاع عليها .

(٣) وظائف قواعد البيانات :

أ. إضافة معلومة أو بيان جديد إلى الملف .

ب. حذف البيانات القديمة والتى لم تعد هناك حاجة إليها .

ت. تغيير بيانات موجودة تبعاً لمعلومات إستحدثت .

ث. البحث والإستعلام عن معلومات محددة .

ج. ترتيب وتنظيم البيانات داخل الملفات .

ح. عرض البيانات فى شكل تقارير أو نماذج منظمة .

خ. حساب المجموع النهائى أو المجموع الفرعى أو المتوسط الحسابى لبيانات مطلوبة .

(٤) أنواع قواعد البيانات :

أ- من حيث الحجم:

١- مشروعات صغيرة

(a) Access (b) Paradox (c)Foxpro (d) DBASE III+/IV
(e)R:BASE

٢- مشروعات كبيرة

(a)Oracle
(b)SQL (Structured Query Language)
(c)DMS (Database Management System)
(d)IDMS (Integrated Database Management System)
(e)Informix
(f)Sybase

ب- من حيث طريقة العمل :

١- قواعد البيانات ذات شكل هرمى Hierarchy Databases

٢- قواعد بيانات شبكية Network Databases

٤-قواعد بيانات علائقية Relational Databases

(٥)تنظيم قواعد البيانات داخل قاعدة البيانات :

تخزن المعلومات المطلوبة لقواعد البيانات داخل ملفات ، وتوضع هذة الملفات على أحد وسائط التخزين المساعدة مثل القرص المغناطيسى .

كل ملف عبارة عن جدول يشتمل على سطور وأعمدة ، ويشتمل كل ملف على مجموعة من السجلات Records ويحتل كل سجل سطرا داخل الملف ، ويقسم كل سجل إلى عدد من الحقول Fields .

إذا أردنا إعداد دليل تليفونات لسكان مدينة القاهرة ، والملف الطلوب يشتمل على البيانات التالية :

رقم المشترك   الأسم   العنوان   رقم الهاتف

جدول البيانات (Database table)

ويحتوى على 3 Records

رقم المشترك	الأسم	العنوان	الهاتف
١-	أيمن عادل نبية	مصر الجديدة	٤٣٣٦٧٥٤
٢-	محمد أحمد منصور	الجيزة	٢٤٨٤٣٥٤
٣=	أحمد محمد مصطفى	الدقى	٢٥٠٥٦٤٣

4 FIELDS

Rec

شكل رقم (١)

يوضح كيفية تنظيم البيانات داخل جدول قاعدة البيانات

 **خطوات تصميم قاعدة البيانات :**

١- تحديد الغرض من قواعد البيانات :

يدلك الغرض من قواعد البيانات على المعلومات التى ستحتاجها منها يمكنك تحديد الموضوعات التى تحتاجها لحفظ حقائق عنها (الجداول) والحقائق المراد حفظها عن كل موضوع (الحقول بداخل الجداول) وفى سبيل تحديد الغرض من أنشاء قاعدة البيانات يمكنك عمل الآتى :-

أ- أن تتحدث مع المستخدمين المرتقبين لقواعد البيانات .

ب- أن تتشاور مع الأخرين حول نوعية الأسئلة التى تود أن تجيب عنها قاعدة البيانات .

ج- أن تضع تخطيطات للتقارير المراد إنتاجها .

د- أن تجمع النماذج المستخدمة بالفعل فى تسجيل البيانات .

وتستعين بكل هذه البيانات فى الخطوات التالية للتصميم .

مثال : متابعة حركة المبيعات والمخزون :

لنفرض أن شركة (جيسكو) للاستيراد والتصدير التى تسوق المواد الغذائية الفاخرة فى جميع انحاء العالم ، تزمع انشاء قاعدة بيانات مبيعات مخزون الشركة.

إبدأ بكتابة قائمة الاسئلة التى يجب أن تجيب عليها قاعدة البيانات . مثل:

١- كم بلغ حجم المبيعات من منتجاتنا المتميزة فى الشهر الماضى؟

٢- اين يقطن أفضل عملاؤنا؟

٣- من هو مورد المنتج الاكثر مبيعا؟

ويمكنك بعد ذلك جمع كل النماذج والتقارير التى تحتوى على المعلومات المزمع ان تكون قاعدة البيانات قادرة على اصدارها ، مع ملاحظة ان الشركة تستخدم حاليا تقريرا مطبوعا لمتابعة المنتجات التى تم طلبها ونموذج طلبيات لتسجيل الطلبيات الجديدة . ويوضح الشكلين التاليين هاتين الوثيقتين .

شكل رقم (٢) يوضح تقرير متابعة المنتجات التى تم طلبها

تقرير متابعة المنتجات التى تم طلبها

٢ سبتمبر ١٩٩٧

إسم الصنف	أسم المنتج	المخزون	المطلوب	إسم المورد	التليفون

شكل رقم (٣) يوضح نموذج الطلبيات

شركو جيسكو للاستيراد والتصدير — نموذج طلبيات

٥٠ شارع الميرغني مصر الجديدة - القاهرة

تليفون :٢٤٥٢٨٤٦ فاكس ٢٤٥٢٨٤٦

الفاتورة إلى : ______________ مشحون إلى ______________

تاريخ الطلبيه : ______________

رقم المنتج	أسم المنتج	سعر [illegible]	الكمية	السعر الإجمالي
			الإجمالي الفرعي	
			تكلفة الشحن	
			الإجمالي	

٢-تحديد الجداول ال لآزمة :

يتضمن نموذج الطلبيات وتقرير متابعة المنتجات معلومات عن الموضوعـــات الأتية :

- العملاء .
- الموردين .
- المنتجات .
- الطلبيات .

وتستطيع من القائمة السابقة الوصول إلى تصور مبدئى حول جـــدول قواعـد البيانات وبعض الحقوق فى كل جدو' .

قاعدة بيانات شركة جيسكو

للإستيراد والتصدير

الموظفين	العملاء	الموردين
الإسم	إسم الشركة	إسم الشركة
العنوان	العنوان	العنوان
	مندوب الشركة	مندوب الشركة
المنتجات	الطلبيات	
إسم المنتج	تاريخ الطلبية	
سعر الوحدة	عنوان المورد	
الوحدات المخزونة		
الوحدات المطلوبة		

شكل (٤)

يوضح تصوير مبدئى لجدول قاعدة البيانات وبعض الحقوق فى كل جدول

٣- تحديد الحقول اللآزمة :

لتحديد الحقول داخل كل جدول ، يجب أن تقرر أولاً ماذا تريد أن تعـرف عـن الأفراد أو الأشياء أو الأحداث المسجلة بالجداول .

- **لمحات إرشادية لتحديد الحقول :**

أ. لاتتضمن بيانات محسوبة أو مستنتجة .

ب. ضمن كل المعلومات التى تحتاجها .

ج. خزين المعلومات حسب أصغر جزء منطقى (أسم المنتج /فئته / وصفة)

- **حقول المفتاح الأساسى :**

أ. لا يسمح بوجود قيما متكررة أو خالية فى حقل المفتاح الأساسى .

ب. قد تستخدم القيمة الموجودة فى حقل المفتاح الأساسى للبحث عن السجلات لذلك يراعـى ألا يحتوى على عدد كبير من الأرقام أو الأحرف .

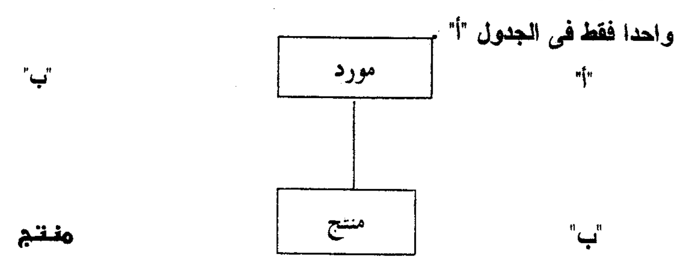
ج- يؤثر حجم المفتاح الأساسى على سرعة العمليات فى قاعدة البيانات .

٤- تحديد العلاقات :

يوجد ثلاثة أنواع من العلاقات بين الجداول :

أ- إنشاء علاقة إرتباط رأس بأطراف (One – to Many)

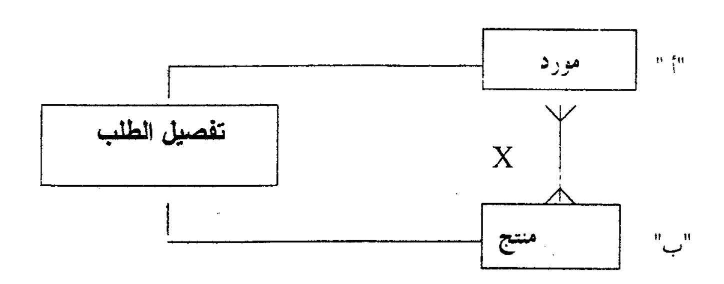
فى هذة يقابل السجل الواحد فى الجدول "أ" عدة سجلات مطابقة فى حقـل معين فى الجدول "ب" ويطابق فى حقل معين أى سجل فى الجـدول "ب" سجلا واحدا فقط فى الجدول "أ" .



شكل (٥) يوضح علاقة إرتباط رأس بأطراف

ب- **إنشاء علاقة إرتباط أطراف بأطراف (Many – to – Many)**

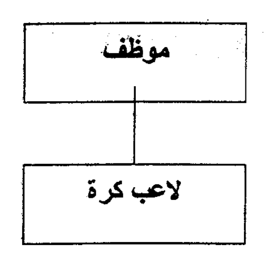
فى هذة العلاقة يقابل السجل الواحد فى الجدول "أ" عدة سجلات فى الجدول "ب" ويقابل السجل الواحد فى الجدول "ب" عدة سجلات فى الجدول "أ"



شكل رقم (٦) يوضح علاقة إرتباط أطراف باطراف

ج- **إنشاء علاقة رأس برأس (One – to – One)**

فى هذا النوع من العلاقات لايقابل أى سجل فى جدول "أ" أكثر من سجل واحد فى جدول "ب" ، ولا يقابل أى سجل فى جدول "ب" أكثر من سجل واحد فى جدول "أ" .



شكل رقم (٧) يوضح علاقة رأس برأس

٥- تنقيح تصميم قواعد البيانات :

تأكد من أنك تستطيع إستخدام قاعدة البيانات للحصول على الإجابات التي تريدها . ثم ضع تخطيطا مبدئياً للنماذج والتقارير وتأكد من أنها ستظهر البيانات المتوقعة منها ثم أبحث عن أى تكرار غير ضرورى للبيانات وتخلص منها .

(٧) بعض الأمثله لقواعدالبيانات :

أ. عناوين العملاء فى دليل العناوين .

ب. معلومات حول مبيعات المنتج فى دفتر الحسابات الجارية .

ت. نماذج معلومات الموظف فى مجلدات الملفات .

وجدير بالذكر أنه مع برنامج Access يمكنك إنشاء قاعدة بيانات علائقية لتخزين كل البيانات المرتبطة بعمل ما . كالبيانات الخاصة بكل من العملاء ، المنتجات ، والموظفين ، وهكذا ...

وتعمل قاعدة البيانات العلائقية على جعل عملية البحث عن بياناتك ، تحليلها ، صيانتها ، وحمايتها أمراً فى غاية السهولة ، حيث يتم تخزين البيانات فى مكان واحد فقط.

قاعدة البيانات ACCESS

مكونات الشاشة الرئيسية لقاعدة البيانات Access

تحتوى قاعدة بيانات ACCESS على ستة أنواع من الكائنات :

١. جداول : تقوم بتخزين بيانات .

٢. إستعلام : تقوم بتجميع البيانات التى تطلبها من جدول أو أكثر ويمكنك عــــرض البيانات أو تحريرها فى نموذج ، كما يمكنك طباعتها وتقرير .

٣. نماذج : تقوم بعرض بيانات من جدول أو إستعلامات ، لذلك فإن بإمكـــــانك إدخال بيانات ، عرضها ، أو تحريرها

٤. تقارير : تقوم بتخليص البيانات وعرضها من الجدول والإستعلانات ، لذلك فــإن بإمكانك طباعة البيانات أو تحليلها .

٥. وحدات ماكرو : تقوم بإتمام قاعدة البيانـــــات الخاصـــة بـــك عـــن طريـــق أداء الإجراءات التى قمت بتحديدها بدون الحاجة إلى البرمجة .

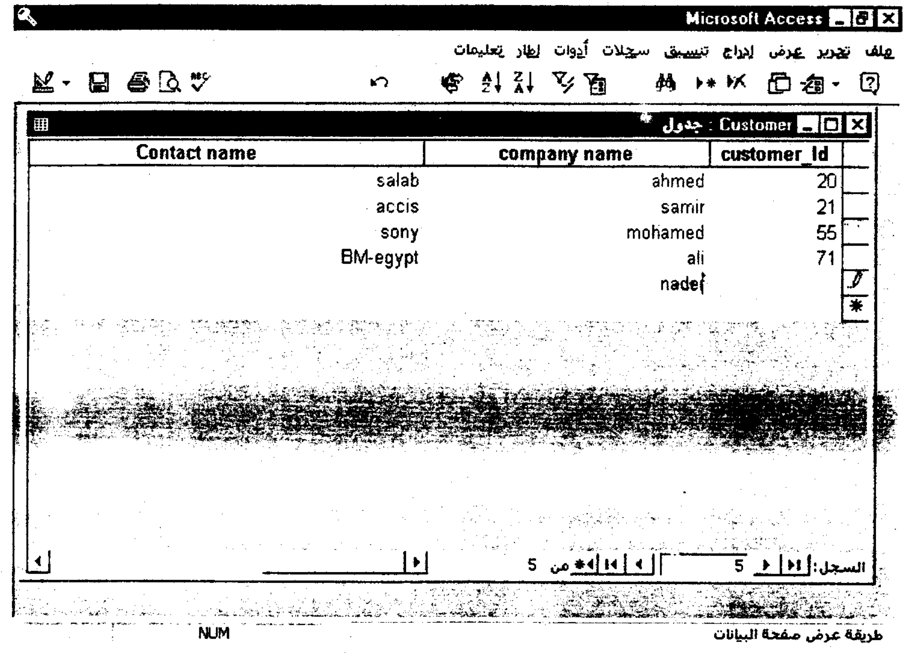
٦. وحدات نمطية : تقوم بتخزين برامج ACCESS Basic الـــــذى يمكنـــك كتابـــة لتخصيص قاعدة بياناتك أو تحسينها أو توسيعها

(١) ماهو الجدول ؟

الجدول هو مجموعة من البيانات حول موضوع محدد . على سبيل المثال ، يمكن أن يحتوى الجدول على بيانات حول Customers(العملاء) .

ينظم الجدول فى أعمدة (تسمى حقول) وصفوف (تسمى سجلات) ، يحتوى كل حقـــل على معلومات حول أحد العملاء مثل Customer ID (رقم العميل)

ويحتوى كل سجل على كل المعلومات المتعلقة بأحد العمـــــلاء متضمناً Customers Id (رقم العميل) Company Name (إسم الشـــركة) ، Contact Name(إســم منـــدوب الشركه)إلخ .



شكل (٩) يوضح جدول بيانات العملاء فى طريقة عرض صفحة البيانات

طريقة عرض الجدول :

- يمكنك العمل بالجدول بطريقتى عرض .

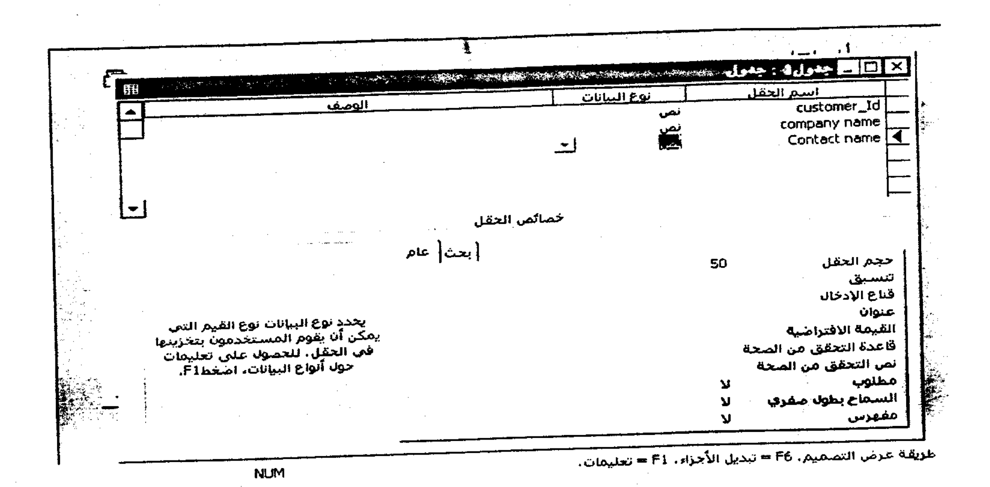
- ***طريقة عرض التصميم أو طريقة عرض صفحة البيانات .***

١- عـرض التصميـم :

لإنشاء أو تحديث هيكل الجدول ، قم بالعمل فى طريقـــة العـــرض تصميـــم الجدول يمكنك تحديد أنواع البيانات التى سوف يحتفظ بها الجدول – مثـــلاً إســـم الموظف محفوظ كنص ، وتارخ التعيين محفوظ كتارخ .

٢- عرض صفحـة البيانـات :

لإضافة ، تحرير ـو تحليل البيانات نفسها – سرد بـالموظفين وتواريـخ التعيين – يتم العمل فى طريقة العرض صفحة صفحة بيانات الجدول .



شكل (١٠)

الجزء العلوى يوضح طريقة عرض تصميم جدول البيانات

والجزء السفلى يوضح طريقة عرض صفحة البيانات

(٢) مـا هو الإسـتعلام ؟

إن الإستعلام ما هو إلا سؤال تسأله عن بياناتك . فأنت قد تسأل مثلاً :

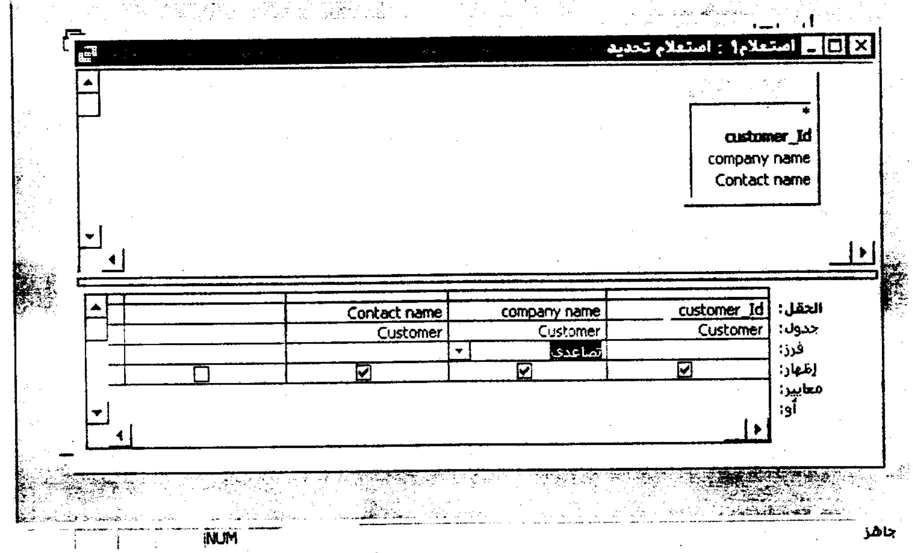
أ- أى الطلبات جاء من العملاء المقيمين فى باريس ؟

ب- من الذى كان أفضل موظف مبيعات خلال الربع الأخير من السنة ؟

ج- ماهى الحسابات المستحقة ؟

يقوم ACCESS بجمع البيانات التى تجيب على سؤالك من جدول أو أكـثر . إن هـذة البيانات هى المجموعة الحيوية (إذا كنت تستطيع التحرير فيها) ، أو لقطة (إذا كنت لاتسـتطيع التحرير فيها)

وفى كل مرة تشغل فيها الإستعلام تحصل على أحدث المعلومات فى المجموعة الحيويه . ويقوم ACCESS إما بعرض المجموعة الحيوية أو اللقطة لمشاهدتها ، أو تنفيذ إجراءما فيـها مثل حذف بعض البيانات أو تحديثها .



شكل(١١)

الجزء العلوى يوضح طريقة عرض تصميم الإستعلام

والجزء السفلى يوضح طريقة عرض صفحة البيانات

طرق عرض الإستعلام :

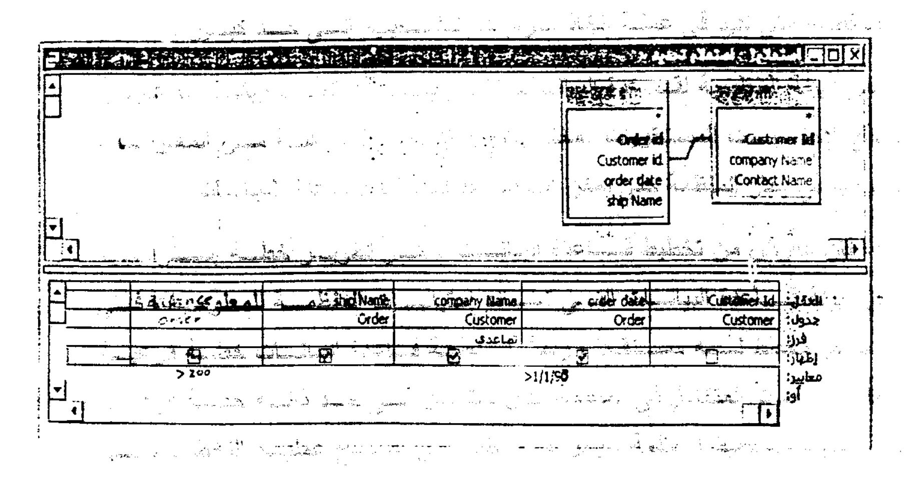
يمكنك العمل بالإستعلام فى طريقتى العرض –تصميم أو صفحة البيانات .

١- عـرض التصميـم :

لإنشاء أو تحديث هيكل الإستعلام ، قم بالعمل فى طريقة العرض تصميم ، هنا تسأل أسئلة عن بياناتك لتحديد لأى بيانات تريدها ، وتحديد كيفية ترتيبها .

٢- عـرض صفحـة بيانـات

لإضافة، تحرير، تحليل البيانات نفسها – البيانات المحتواة بداخل المجموعة الاحصائية أو نفثة السجل التى تجيب عن الأسئلة التى إستوحتها – قم بالعمل فى عرض إستعلام البيانات .



شكل (١٢) يوضح الإستعلام عن طلبات العملاء
بعد يوم ١٩٩٣/١/١ وأكثر من ٢٠٠ دولار

(٣)ماهو النموذج ؟

يستخدم النموذج فى عرض وتحرير المعلومات فى قاعدة البيانات سجلاً سجلاً .

النمـوذج :

١. يعرض فـقط المعلومات التى تريد أن ترها وبالطريقة التى تريد أن تراها بها .
٢. يستخدم عناصر تحكم مألوفة مثل مربعات النصوص وخانات الإختيار المســتخدمة فـى Windows مما يجعل عرض وإدخال البيانات سهلاً .
٣. يمكن أن يكون ملون ومميز الشكل لأن لك إمكانيات التحكم فى حجم ومظــهر كــل مــن مكوناته.

Customer

customer_Id 20

company name ahmed

Contact name salab

السجل: 1 من 5

شكل (١٣) يوضح طريقة عرض نموذج البيانات

طرق عرض النموذج :

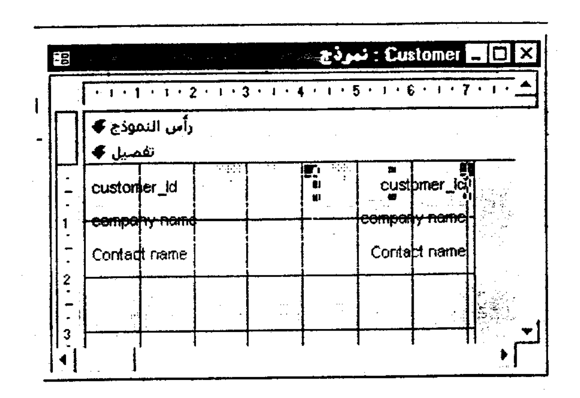
يمكنك العمل بالنموذج فى طرق عرض مختلفة – طريقة عرض التصميم الأساسى وطريقة عرض النموذج .

١- عرض التصميم :

لإنشاء أو تحديث هيكل النموذج ، يتم العمل فى تصميم النموذج ، إضافة التحكمات Controls) إلى النموذج المنضمين إلى الحقوق فى الجدول أو الإستعلام ، بما فيهم مربعات النص ، أزرار الإختيارات ، الرسومات ، والصور .

٢- عرض النموذج :

لإضافة ،تحرير أو تحليل البيانات نفسها ، سجل سجل ، يتم العمل فى عرض النموذج . يمكنك أيضاً العمل بالنموذج فى عرض قاعدة البيانات لمشاهدة عرض تقديمى صف وأعمدة للبيانات .



شكل (١٤) يوضح طريقة عرض تصميم نموذج البيانات

(٤) ماهو التقرير ؟

التقريــر :

١- يعرض المعلومات التى تريدها فقط بالطريقة التى تريدها .

٢- يمكنه تجميع السجلات فى عــدة مســتويات ، كمــا يمكنــه حســاب الأجماليــات والمتوسطات بإختيار القيم من عدة سجلات ، كذلك فإنه جذاب ومميز حيـــث أنــك تملك القدرة على التحكم فى حجم كل شئ فى التقرير ومظهره .

طرق عــرض التقريــر ؟

يمكنك العمل بالتقرير فى طريقتى العرض – تصميم ومعاينة قبل الطباعة .

١- عرض التصميـــم :

لإنشاء تصميم أو تحديث هيكل التقرير ، يتم العمل فى عرض التصميـــم . يمكنك إضافة عناصر التحكم التى تنضم إلى الحقوق فـــى الجــدول أو عنـــاصر التحكم غير المنضمة التى تحسب الإجماليات أو المتوسطات

٢- معاينة قبل الطباعة :

لطباعة أو تحليل البيانات نفسها ، يتم العمل فى المعاينه قبل الطباعة ، يمكنـــك أيضاً فتح التقرير فى عرض تمهيدى للمثال للتأكد بسرعة من تخطيطها .

(٥) مـا هو المــاكرو ؟

الماكرو هي مجموعة من الإجراءات كل إجراء فى الماكرو يفعل شـــئ –مثـل فتـح النموذج أو طباعة التقرير . بإمكانك كتابة وحدات الماكرو ولتشغيل المهام الشائعة تلقائيـاً – لمساعدتك فى العمل بكفاءة وتوفير الوقت .

إذا أديت مهام معينه بصفه متكررة ، فمن الأرجح إنشاء ماكروينفذها لك .

بعض الأشياء التىيمكن أن تقوم وحدات الماكرو :

١. فتح النماذج التى تستخدمها يومياً تلقائياً .
٢. فتح النموذج الثانى بنقرة زر ، والبحث عن السجلات المرتبطة بهذا النموذج .
٣. إعداد وطباعة التقارير .
٤. مراجعة البيانات للتأكد من صحتها .

(٥) ماهى الوحدات النمطية ؟

الوحدات النمطية وحدات من الرمز كتبت بلغة Access Basic . يمكنـك كتابـة وإستعمال الوحدات النمطية للتحول التلقائى وتخصيص قاعدة بياناتك بطرق معقدة جداً.

ولإستعمال الوحدات النمطية تحتاج لأن تكون لديك دراية بالبرمجة بلغة البيســك ، أو مستعداً للتعلم . وهى أقوى بكثير من وحدات الماكرو ، ولكن أكثر تعقيداً فى الكتابة .

وفيما يلى نعرض أحد التطبيقات المدعمه لبرنامج قاعدة البيانات Access والذى يعرف بقاعدة البيانات Northwind:

Microsoft Access

ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me

Northwind : قاعدة بيانات

جداول استعلامات نماذج تقارير وحدات ماكرو وحدات نمطية

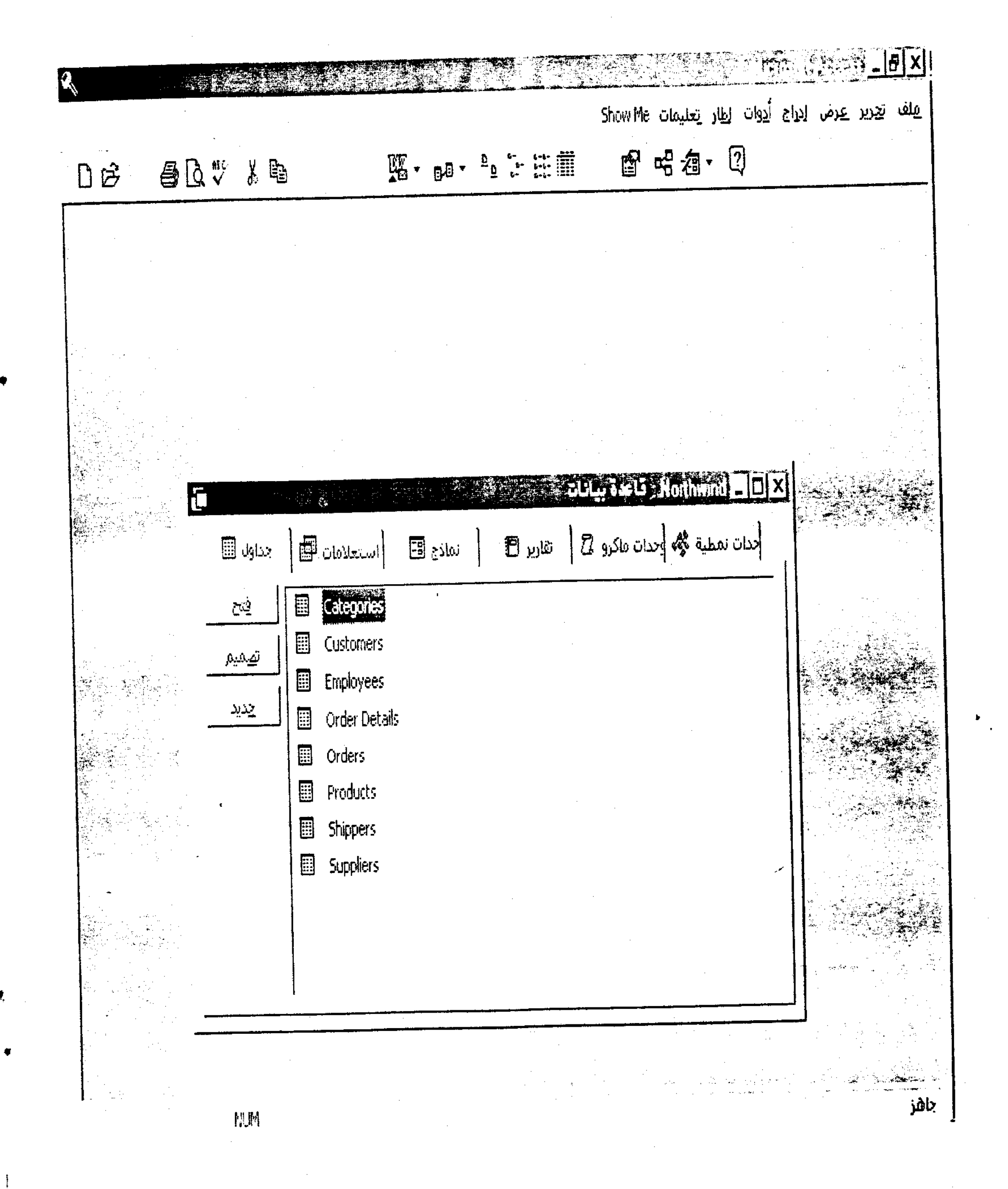
فتح

تصميم

جديد

Categories
Customers
Employees
Order Details
Orders
Products
Shippers
Suppliers

جاهز NUM


ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
قاعدة بيانات : Northwind
جداول
استعلامات
نماذج
تقارير
وحدات ماكرو
وحدات نمطية
فتح
تصميم
جديد
Categories
Customers
Employees
Order Details
Orders
Products
Shippers
Suppliers
جاهز
NUM

ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me

اسم الحقل	نوع البيانات	الوصف
CategoryID	ترقيم تلقائي	Number automatically assigned to a new category.
CategoryName	نص	Name of food category.
Description	مذكرة	
Picture	كائن OLE	A picture representing the food category.

خصائص الحقل

عام | بحث

حجم الحقل	عدد صحيح طويل
القيم الجديدة	زيادة
تنسيق	
عنوان	Category ID
مفهرس	نعم (بدون تكرار)

يمكن أن يصل طول اسم الحقل إلى ٦٤ حرفاً تتضمن مسافات. للحصول على تعليمات حول أسماء الحقول اضغط F1.

طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات.

NUM

ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me

اسم الحقل	نوع البيانات	الوصف
CategoryID	ترقيم تلقائي	Number automatically assigned to a new category.
CategoryName	نص	Name of food category.
Description	مذكرة	
Picture	كائن OLE	A picture representing the food category.

خصائص الحقل

عام | بحث

حجم الحقل	15
تنسيق	
قناع الإدخال	
عنوان	Category Name
القيمة الافتراضية	
قاعدة التحقق من الصحة	
نص التحقق من الصحة	
مطلوب	نعم
السماح بطول صفري	لا
مفهرس	نعم (بدون تكرار)

يمكن أن يصل طول اسم الحقل إلى ٦٤ حرفاً تتضمن مسافات. للحصول على تعليمات حول أسماء الحقول، اضغط F1.

طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات.

NUM

ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me

اسم الحقل	نوع البيانات	الوصف
CustomerID	نص	Unique five-character code based on customer name.
CompanyName	نص	
ContactName	نص	
ContactTitle	نص	
Address	نص	Street or post-office box.
City	نص	
Region	نص	State or province.
PostalCode	نص	
Country	نص	
Phone	نص	Phone number includes country code or area code.
Fax	نص	Phone number includes country code or area code.

خصائص الحقل

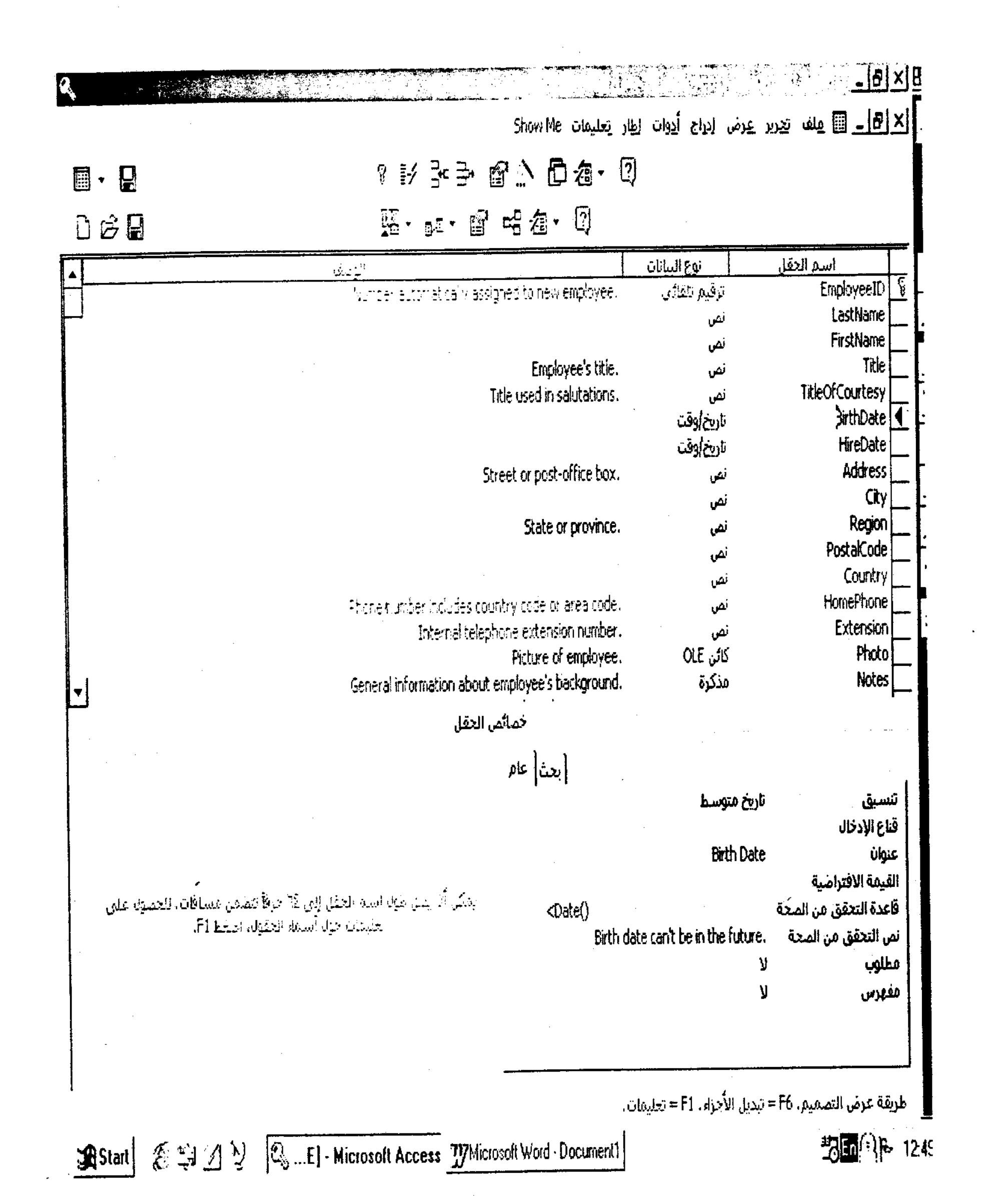
عام | بحث

حجم الحقل	5
تنسيق	
قناع الإدخال	>LLLLL
عنوان	Customer ID
القيمة الافتراضية	
قاعدة التحقق من الصحة	
نص التحقق من الصحة	
مطلوب	لا
السماح بطول صفري	لا
مفهرس	نعم (بدون تكرار)

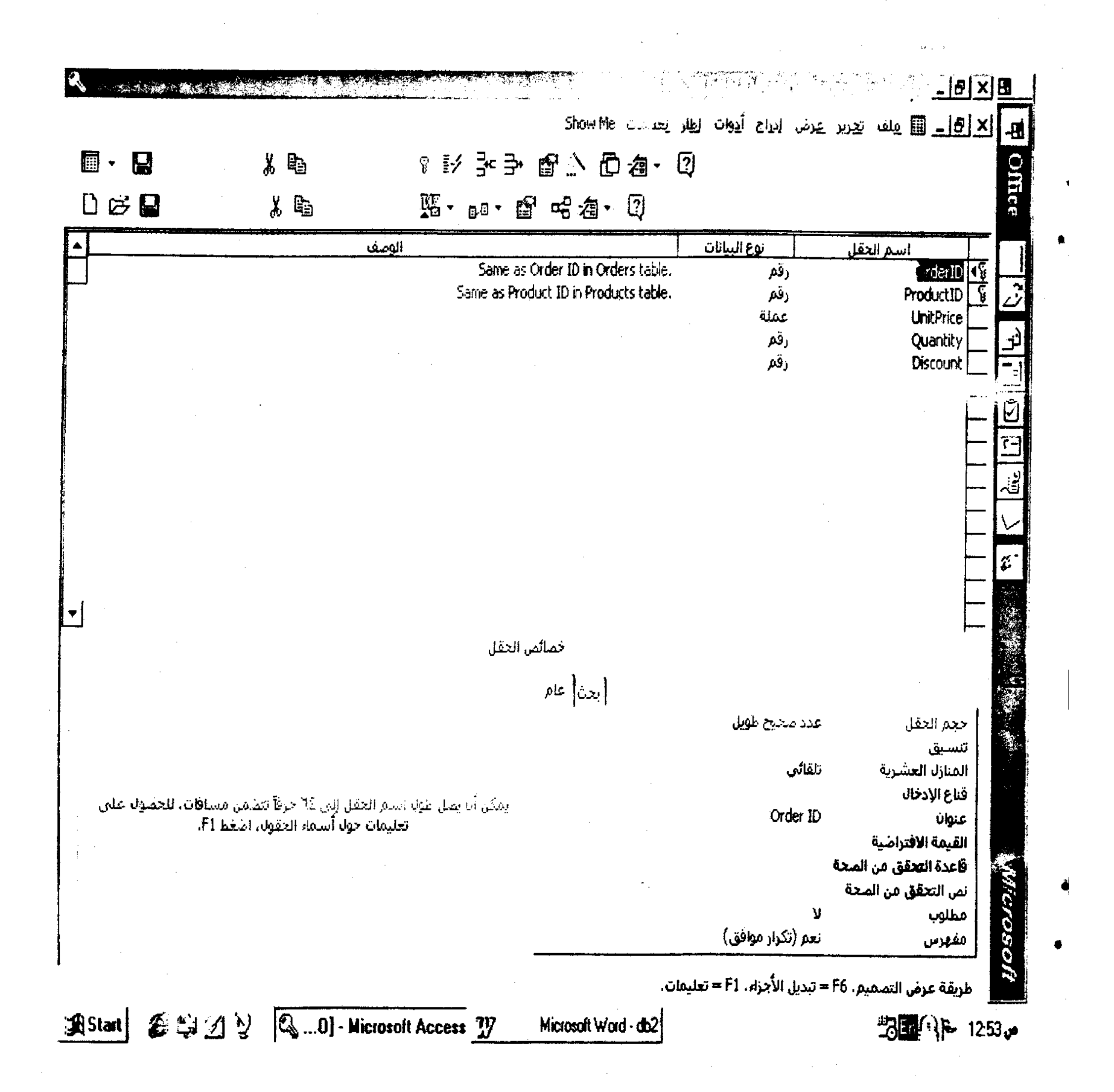
يمكن أن يصل طول اسم الحقل إلى 64 حرفاً تتضمن مسافات. للحصول على تعليمات حول أسماء الحقول اضغط F1.

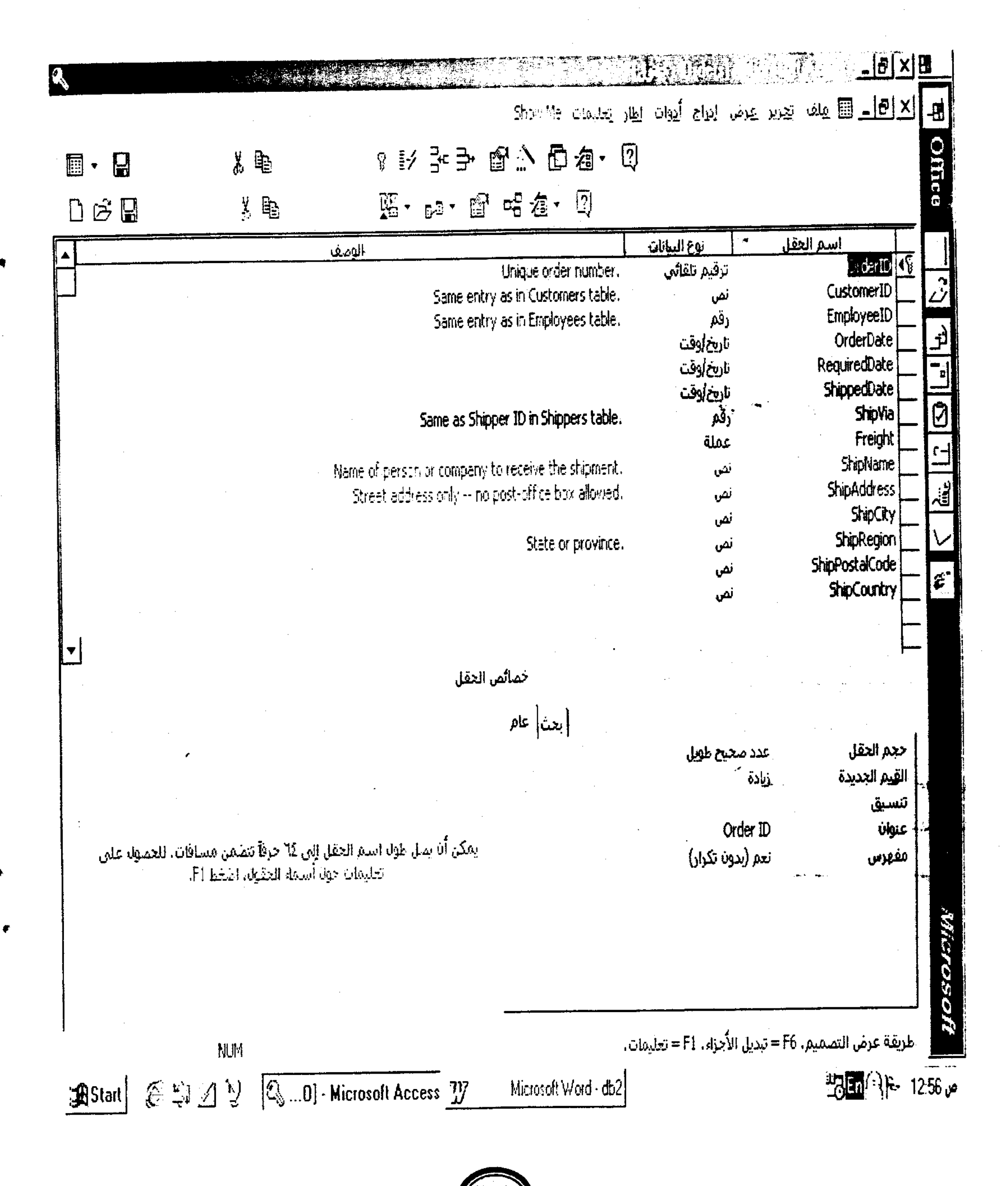
طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات.

NUM



ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
اسم الحقل
نوع البيانات
EmployeeID
ترقيم تلقائي
Number automatically assigned to new employee.
LastName
نص
FirstName
نص
Title
نص
Employee's title.
TitleOfCourtesy
نص
Title used in salutations.
BirthDate
تاريخ/وقت
HireDate
تاريخ/وقت
Address
نص
Street or post-office box.
City
نص
Region
نص
State or province.
PostalCode
نص
Country
نص
HomePhone
نص
Phone number includes country code or area code.
Extension
نص
Internal telephone extension number.
Photo
كائن OLE
Picture of employee.
Notes
مذكرة
General information about employee's background.
خصائص الحقل
عام
بحث
تنسيق
تاريخ متوسط
قناع الإدخال
عنوان
Birth Date
القيمة الافتراضية
قاعدة التحقق من الصحة
<Date()
نص التحقق من الصحة
Birth date can't be in the future.
مطلوب
لا
مفهرس
لا
طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات.
Start
Microsoft Access
Microsoft Word - Document1
12:49


ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
اسم الحقل
نوع البيانات
الوصف
OrderID
رقم
Same as Order ID in Orders table.
ProductID
رقم
Same as Product ID in Products table.
UnitPrice
عملة
Quantity
رقم
Discount
رقم
خصائص الحقل
عام
بحث
حجم الحقل
عدد صحيح طويل
تنسيق
المنازل العشرية
تلقائي
قناع الإدخال
عنوان
Order ID
القيمة الافتراضية
قاعدة التحقق من الصحة
نص التحقق من الصحة
مطلوب
لا
مفهرس
نعم (تكرار موافق)
تعليمات حول أسماء الحقول، اضغط F1.
طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات.
Start
Microsoft Access
Microsoft Word - db2
12:53
Office
Microsoft



ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
اسم الحقل
نوع البيانات
الوصف
OrderID
ترقيم تلقائي
Unique order number.
CustomerID
نص
Same entry as in Customers table.
EmployeeID
رقم
Same entry as in Employees table.
OrderDate
تاريخ/وقت
RequiredDate
تاريخ/وقت
ShippedDate
تاريخ/وقت
ShipVia
رقم
Same as Shipper ID in Shippers table.
Freight
عملة
ShipName
نص
Name of person or company to receive the shipment.
ShipAddress
نص
Street address only -- no post-office box allowed.
ShipCity
نص
ShipRegion
نص
State or province.
ShipPostalCode
نص
ShipCountry
نص
خصائص الحقل
عام
بحث
حجم الحقل
عدد صحيح طويل
القيم الجديدة
زيادة
تنسيق
عنوان
Order ID
مفهرس
نعم (بدون تكرار)
يمكن أن يصل طول اسم الحقل إلى ٦٤ حرفاً تتضمن مسافات. للحصول على تعليمات حول أسماء الحقول، اضغط F1.
طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات.
NUM
Start
Microsoft Access
Microsoft Word - db2
12:56 ص
Office
Microsoft

ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me

اسم الحقل	نوع البيانات	الوصف
ProductID	ترقيم تلقائي	Number automatically assigned to new product.
ProductName	نص	
SupplierID	رقم	Same entry as in Suppliers table.
CategoryID	رقم	Same entry as in Categories table.
QuantityPerUnit	نص	(e.g., 24-count case, 1-liter bottle).
UnitPrice	عملة	
UnitsInStock	رقم	
UnitsOnOrder	رقم	
ReorderLevel	رقم	Minimum units to maintain in stock.
Discontinued	نعم/لا	Yes means item is no longer available.

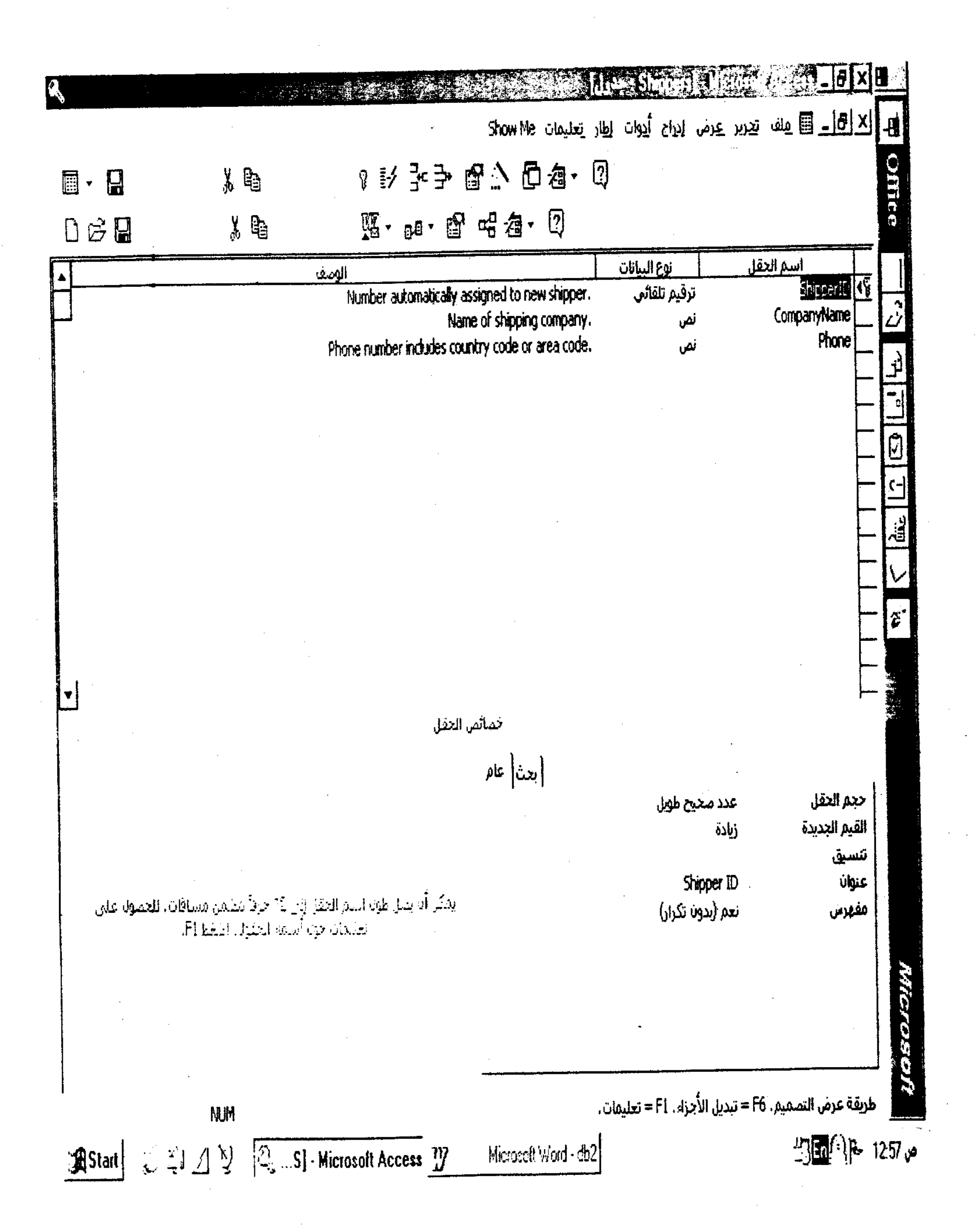
خصائص الحقل

عام | بحث

حجم الحقل	عدد صحيح طويل
القيم الجديدة	زيادة
تنسيق	
عنوان	Product ID
مفهرس	نعم (بدون تكرار)

يمكن أن يصل طول اسم الحقل إلى ٦٤ حرفاً تتضمن مسافات. للحصول على تعليمات حول أسماء الحقول، اضغط F1.

طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات. NUM



ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
اسم الحقل
نوع البيانات
الوصف
ShipperID
ترقيم تلقائي
Number automatically assigned to new shipper.
CompanyName
نص
Name of shipping company.
Phone
نص
Phone number includes country code or area code.
خصائص الحقل
عام
بحث
حجم الحقل
عدد صحيح طويل
القيم الجديدة
زيادة
تنسيق
عنوان
Shipper ID
مفهرس
نعم (بدون تكرار)
طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات.
NUM
Start
...S] - Microsoft Access
Microsoft Word - db2
12:57 م
Office
Microsoft

Microsoft Access - [Suppliers : جدول]

ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me

اسم الحقل	نوع البيانات	الوصف
SupplierID	ترقيم تلقائي	Number automatically assigned to new supplier.
CompanyName	نص	
ContactName	نص	
ContactTitle	نص	
Address	نص	Street or post-office box.
City	نص	
Region	نص	State or province.
PostalCode	نص	
Country	نص	
Phone	نص	Phone number includes country code or area code.
Fax	نص	Phone number includes country code or area code.
HomePage	ارتباط تشعبي	Supplier's home page on World Wide Web.

خصائص الحقل

عام | بحث

حجم الحقل	عدد صحيح طويل
القيم الجديدة	زيادة
تنسيق	
عنوان	Supplier ID
مفهرس	نعم (بدون تكرار)

يمكن أن يصل طول اسم الحقل إلى ٦٤ حرفاً تتضمن مسافات. للحصول على تعليمات حول أسماء الحقول، اضغط F1.

طريقة عرض التصميم. F6 = تبديل الأجزاء. F1 = تعليمات. NUM

Show Me تعليمات إطار أدوات سجلات تنسيق إدراج عرض تحرير ملف

CustomerID ▾ Arial (Arabic) ▾ 10 ▾ B I U

Customer ID	Company Name	Contact Name	Contact Title
ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative
ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner
ANTON	Antonio Moreno Taquería	Antonio Moreno	Owner
AROUT	Around the Horn	Thomas Hardy	Sales Representative
BERGS	Berglunds snabbk.p	Christina Berglund	Order Administrator
BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative
BLONP	Blondel père et fils	Frédérique Citeaux	Marketing Manager
BOLID	B'lido Comidas preparadas	Martín Sommer	Owner
BONAP	Bon app'	Laurence Lebihan	Owner
BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Manager
BSBEV	B's Beverages	Victoria Ashworth	Sales Representative
CACTU	Cactus Comidas para llevar	Patricio Simpson	Sales Agent
CENTC	Centro comercial Moctezuma	Francisco Chang	Marketing Manager
CHOPS	Chop-suey Chinese	Yang Wang	Owner
COMMI	Comércio Mineiro	Pedro Afonso	Sales Associate
CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative
DRACD	Drachenblut Delikatessen	Sven Ottlieb	Order Administrator
DUMON	Du monde entier	Janine Labrune	Owner
EASTC	Eastern Connection	Ann Devon	Sales Agent
ERNSH	Ernst Handel	Roland Mendel	Sales Manager
FAMIA	Familia Arquibaldo	Aria Cruz	Marketing Assistant
FISSA	FISSA Fabrica Inter. Salchichas S.A.	Diego Roel	Accounting Manager
FOLIG	Folies gourmandes	Martine Rancé	Assistant Sales Agent

السجل: 1 من 91

NUM Unique five-character code based on customer name.

Start Microsoft Word - db2.doc ...C] - Microsoft Access 12:11 ص

Microsoft Office

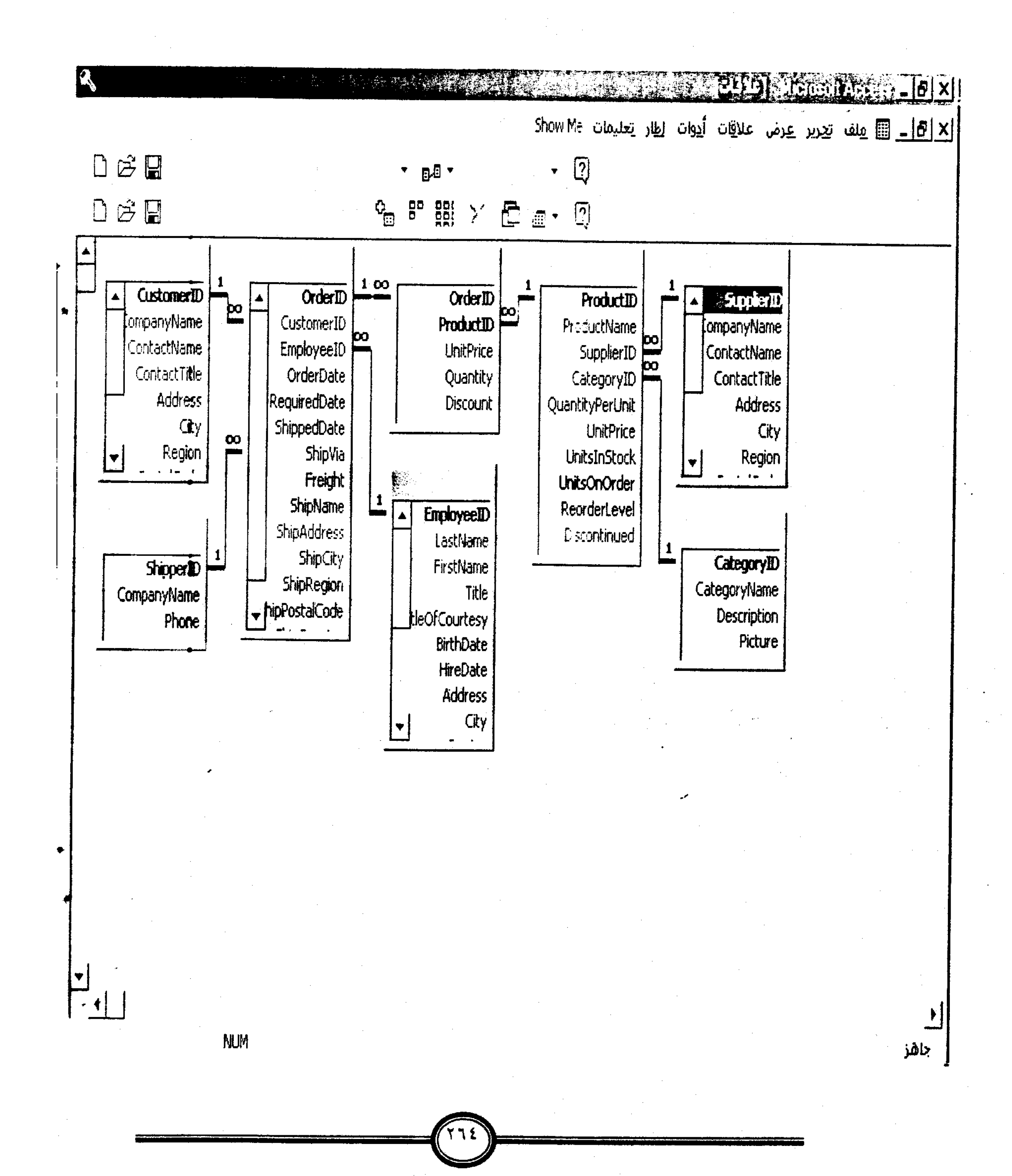
ملف تحرير عرض إدراج تنسيق سجلات أدوات إطار تعليمات Show Me

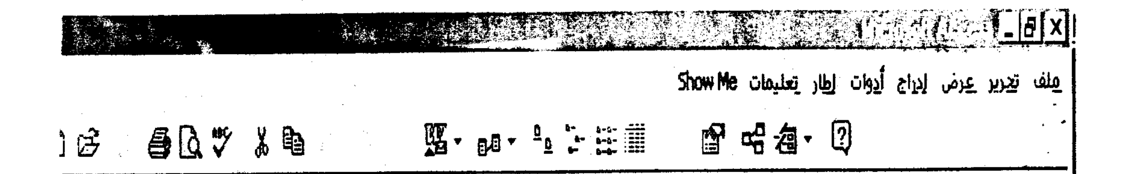
Address · Arial (Arabic) · 10 · B I U

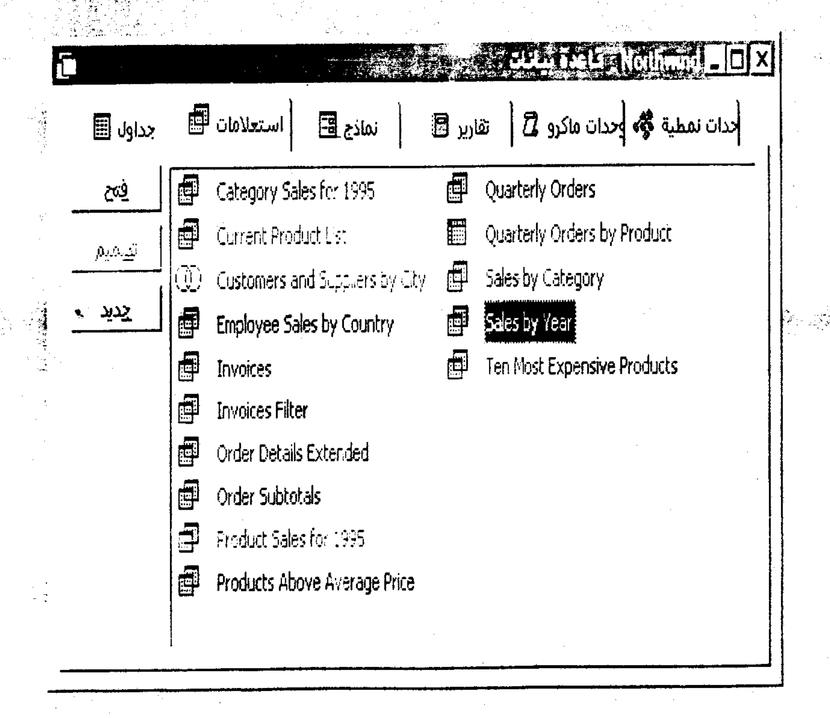
Address	City	Region	Postal Code	Country	Phone
Obere Str. 57	Berlin		12209	Germany	030-0074321
Avda. de la Constitución 2222	México D.F.		05021	Mexico	(5) 555-4729
Mataderos 2312	México D.F.		05023	Mexico	(5) 555-3932
120 Hanover Sq.	London		WA1 1DP	UK	(171) 555-7788
Berguvsvägen 8	Luleå		S-958 22	Sweden	0921-12 34 65
Forsterstr. 57	Mannheim		68306	Germany	0621-08460
24, place Kléber	Strasbourg		67000	France	88.60.15.31
C/ Araquil, 67	Madrid		28023	Spain	(91) 555 22 82
12, rue des Bouchers	Marseille		13008	France	91.24.45.40
23 Tsawassen Blvd.	Tsawassen	BC	T2F 8M4	Canada	(604) 555-4729
Fauntleroy Circus	London		EC2 5NT	UK	(171) 555-1212
Cerrito 333	Buenos Aires		1010	Argentina	(1) 135-5555
Sierras de Granada 9993	México D.F.		05022	Mexico	(5) 555-3392
Hauptstr. 29	Bern		3012	Switzerland	0452-076545
Av. dos Lusíadas, 23	São Paulo	SP	05432-043	Brazil	(11) 555-7647
Berkeley Gardens	London		WX1 6LT	UK	(171) 555-2282
Walserweg 21	Aachen		52066	Germany	0241-039123
67, rue des Cinquante Otages	Nantes		44000	France	40.67.88.88
35 King George	London		WX3 6FW	UK	(171) 555-0297
Kirchgasse 6	Graz		8010	Austria	7675-3425
Rua Orós, 92	São Paulo	SP	05442-030	Brazil	(11) 555-9857
C/ Moralzarzal, 86	Madrid		28034	Spain	(91) 555 94 44
184, chaussée de Tournai	Lille		59000	France	20.16.10.16

السجل: 1 من 91

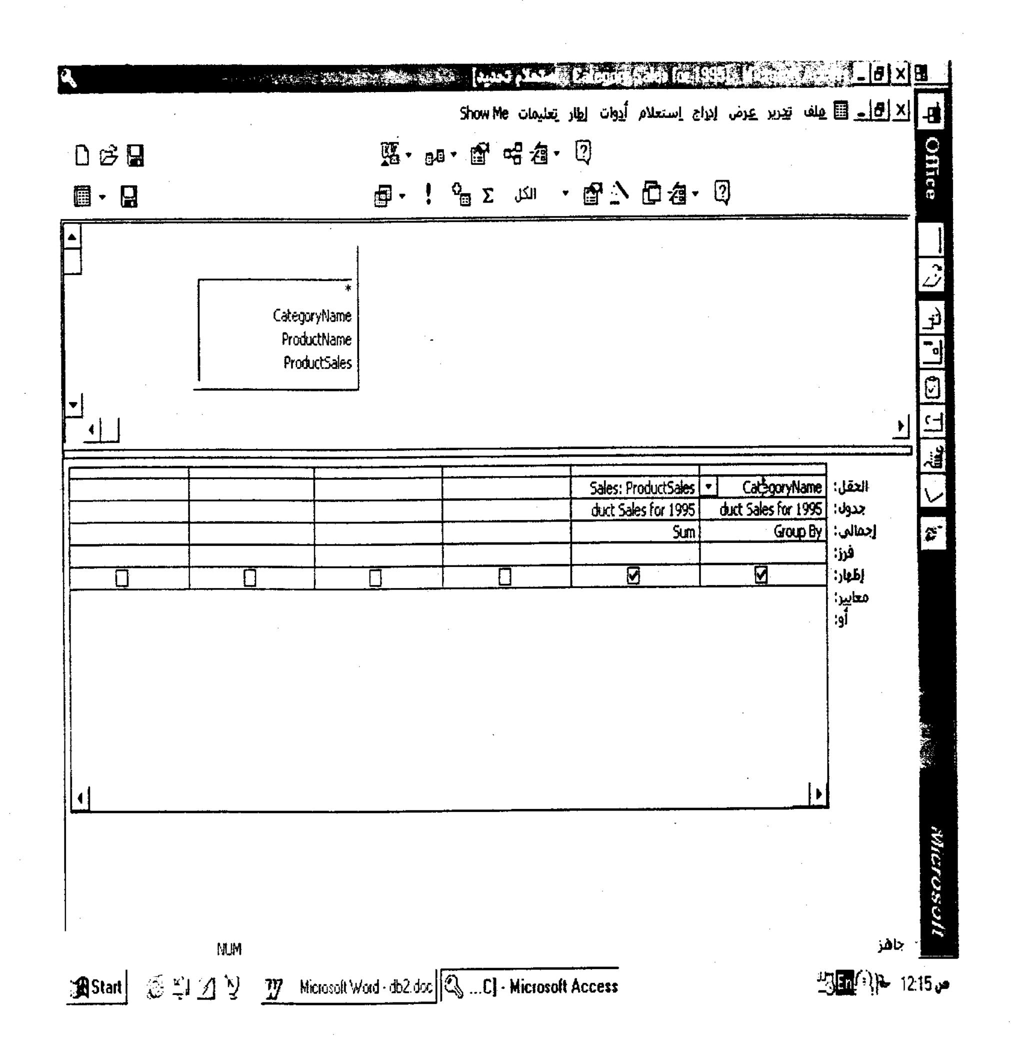
NUM Street or post-office box.


Microsoft Access - [علاقات]
ملف تحرير عرض علاقات أدوات إطار تعليمات Show Me
CustomerID
CompanyName
ContactName
ContactTitle
Address
City
Region
ShipperID
CompanyName
Phone
OrderID
CustomerID
EmployeeID
OrderDate
RequiredDate
ShippedDate
ShipVia
Freight
ShipName
ShipAddress
ShipCity
ShipRegion
ShipPostalCode
OrderID
ProductID
UnitPrice
Quantity
Discount
EmployeeID
LastName
FirstName
Title
TitleOfCourtesy
BirthDate
HireDate
Address
City
ProductID
ProductName
SupplierID
CategoryID
QuantityPerUnit
UnitPrice
UnitsInStock
UnitsOnOrder
ReorderLevel
Discontinued
SupplierID
CompanyName
ContactName
ContactTitle
Address
City
Region
CategoryID
CategoryName
Description
Picture
NUM
جاهز

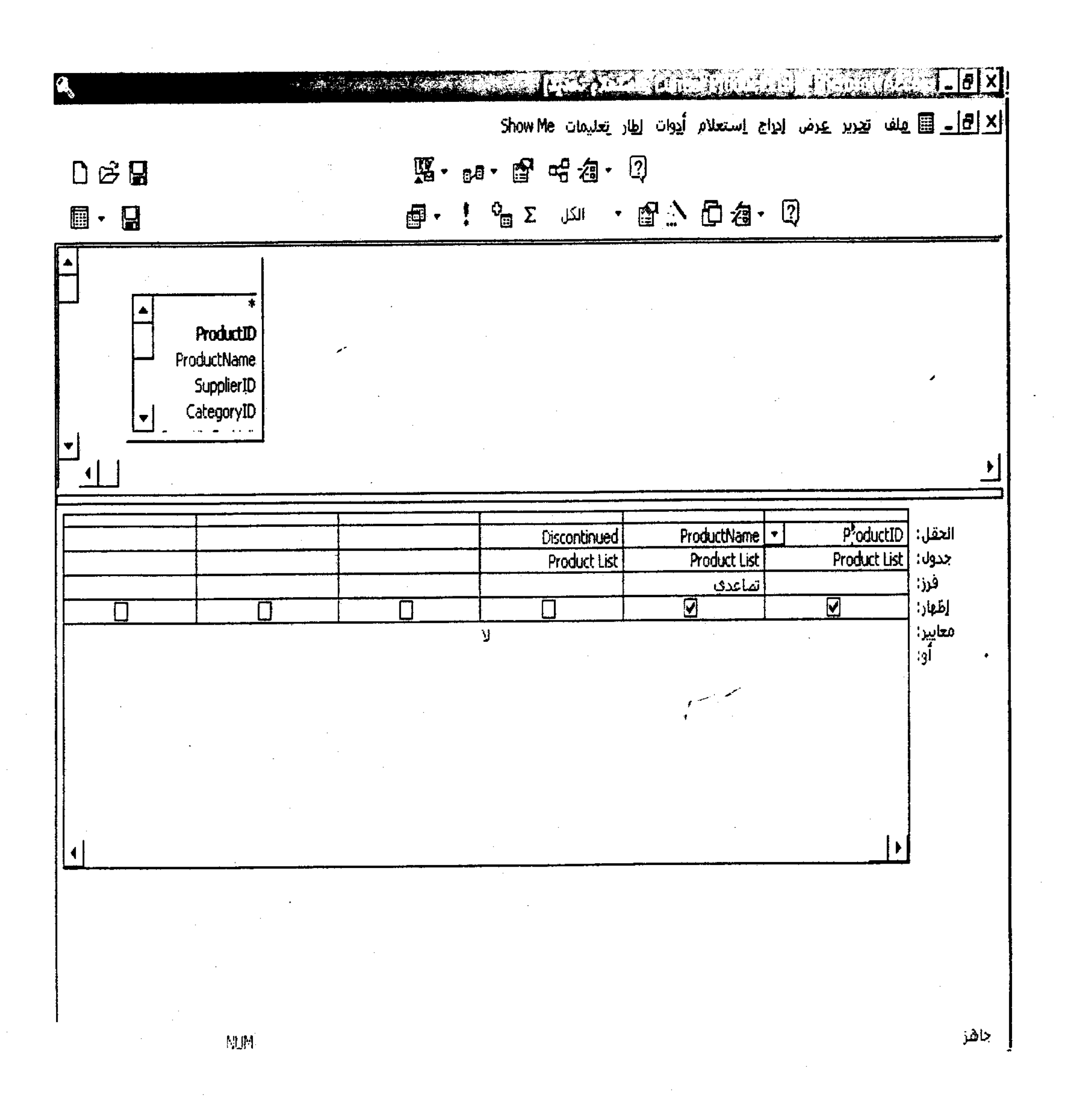

ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me


Northwind : قاعدة بيانات
جداول
استعلامات
نماذج
تقارير
وحدات ماكرو
وحدات نمطية
فتح
تصميم
جديد
Category Sales for 1995
Current Product List
Customers and Suppliers by City
Employee Sales by Country
Invoices
Invoices Filter
Order Details Extended
Order Subtotals
Product Sales for 1995
Products Above Average Price
Quarterly Orders
Quarterly Orders by Product
Sales by Category
Sales by Year
Ten Most Expensive Products

جاهز NUM



Category Name	Category Sales
Beverages	١٠٤,٧٣٧,٦٨ ر.س
Condiments	٥٠,٩٥٢,٦٠ ر.س
Confections	٧٨,١٢٨,٧٣ ر.س
Dairy Products	١١٧,٧٩٧,١٧ ر.س
Grains/Cereals	٥٢,٩٠٢,٣٧ ر.س
Meat/Poultry	٨٠,١٦٠,١٥ ر.س
Produce	٤٧,٤٩١,٥٦ ر.س
Seafood	٦٢,٤٣٥,٠٢ ر.س


ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
الكل
ProductID
ProductName
SupplierID
CategoryID
الحقل: ProductID ProductName Discontinued
جدول: Product List Product List Product List
فرز: تصاعدي
إظهار:
معايير: لا
أو:
جاهز
NUM

	Product ID	Product Name
▶	3	Aniseed Syrup
	40	Boston Crab Meat
	60	Camembert Pierrot
	18	Carnarvon Tigers
	1	Chai
	2	Chang
	39	Chartreuse verte
	4	Chef Anton's Cajun Seasoning
	48	Chocolade
	38	Côte de Blaye
	58	Escargots de Bourgogne
	52	Filo Mix
	71	Flotemysost
	33	Geitost
	15	Genen Shouyu
	56	Gnocchi di nonna Alice
	31	Gorgonzola Telino
	6	Grandma's Boysenberry Spread
	37	Gravad lax
	69	Gudbrandsdalsost
	44	Gula Malacca
	26	Gumbär Gummibärchen

السجل: 1 من 69

استعلام توحيد: Customers and Suppliers by City : Microsoft Access

ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me

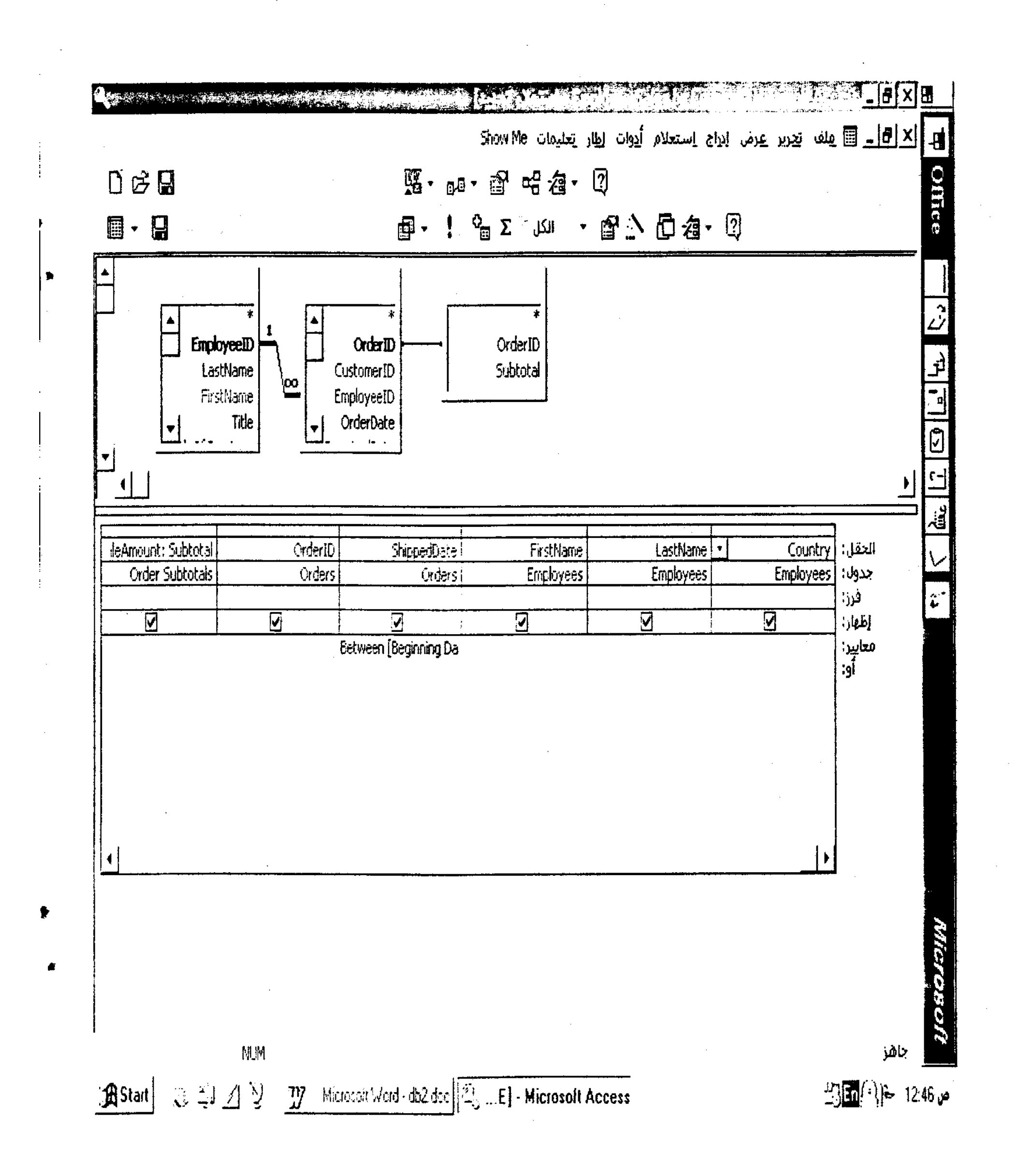
```
SELECT City, CompanyName, ContactName, "Customers" AS [Relationship]
FROM Customers
UNION SELECT City, CompanyName, ContactName, "Suppliers"
FROM Suppliers
ORDER BY City, CompanyName;
```

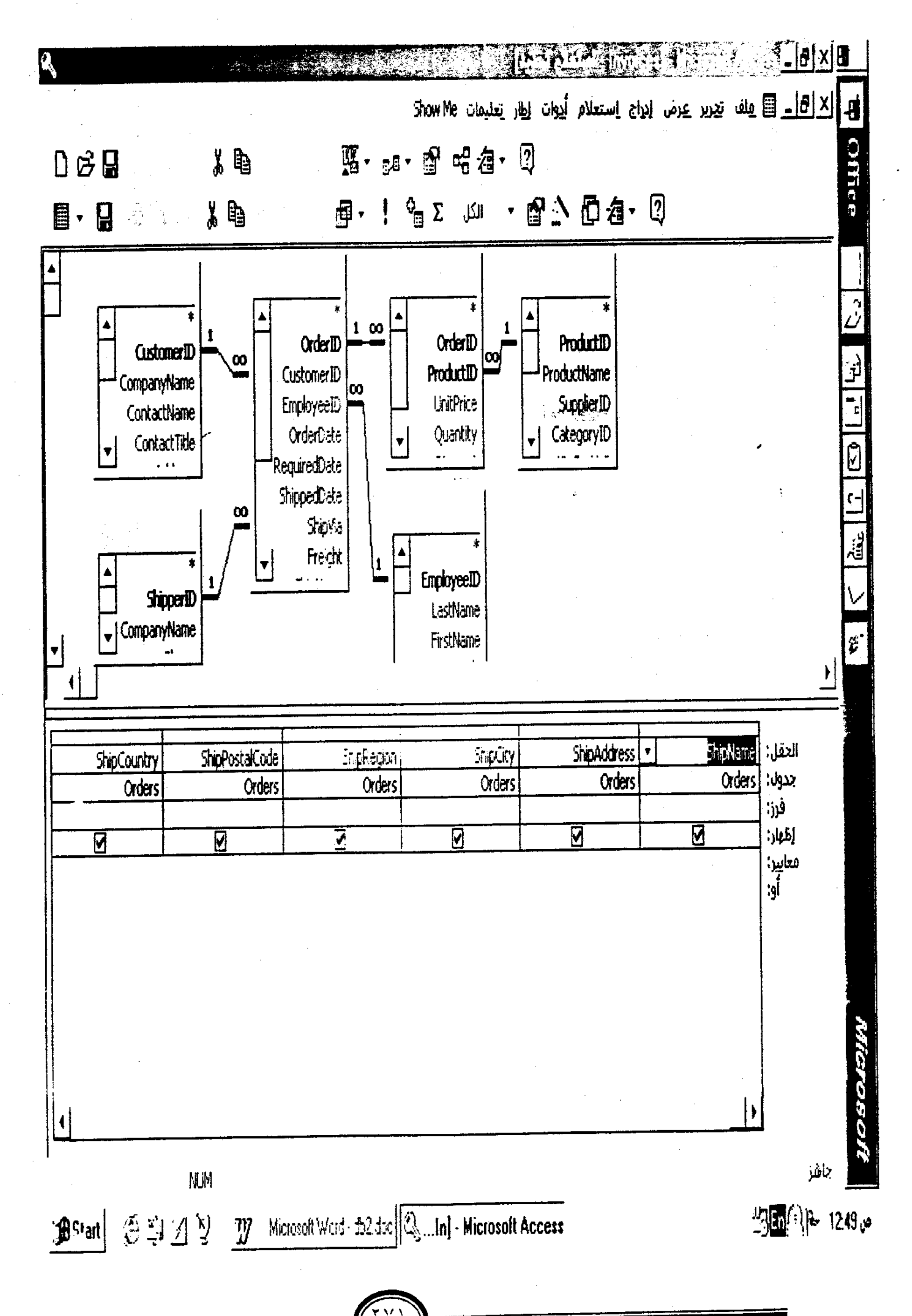
جاهز NUM

City	CompanyName	ContactName	Relationship
Aachen	Drachenblut Delikatessen	Sven Ottlieb	Customers
Albuquerque	Rattlesnake Canyon Grocery	Paula Wilson	Customers
Anchorage	Old World Delicatessen	Rene Phillips	Customers
Ann Arbor	Grandma Kelly's Homestead	Regina Murphy	Suppliers
Annecy	Gai pâturage	Eliane Noz	Suppliers
Barcelona	Galería del gastrónomo	Eduardo Saavedra	Customers
Barquisimeto	LILA-Supermercado	Carlos González	Customers
Bend	Bigfoot Breweries	Cheryl Saylor	Suppliers
Bergamo	Magazzini Alimentari Riuniti	Giovanni Rovelli	Customers
Berlin	Alfreds Futterkiste	Maria Anders	Customers
Berlin	Heli Süßwaren GmbH & Co. KG	Petra Winkler	Suppliers
Bern	Chop-suey Chinese	Yang Wang	Customers
Boise	Save-a-lot Markets	Jose Pavarotti	Customers
Boston	New England Seafood Cannery	Robb Merchant	Suppliers
Brandenburg	Kniglich Essen	Philip Cramer	Customers
Bruxelles	Maison Dewey	Catherine Dewey	Customers
Bräcke	Folk och fä HB	Maria Larsson	Customers
Buenos Aires	Cactus Comidas para llevar	Patricio Simpson	Customers
Buenos Aires	Océano Atlántico Ltda.	Yvonne Moncada	Customers
Buenos Aires	Rancho grande	Sergio Gutiérrez	Customers
Butte	The Cracker Box	Liu Wong	Customers
Campinas	Gourmet Lanchonetes	André Fonseca	Customers

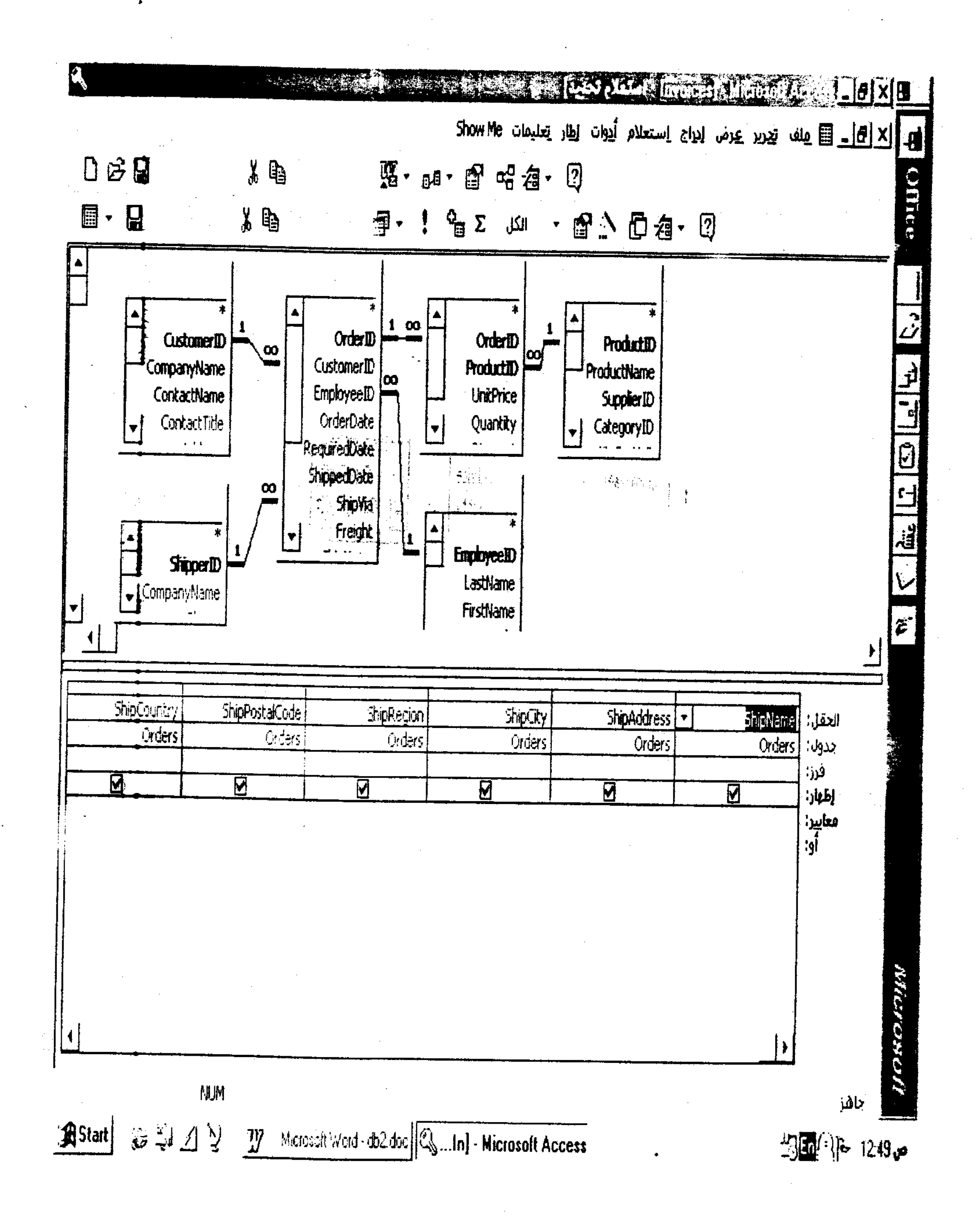
السجل: 1 من 120

NUM

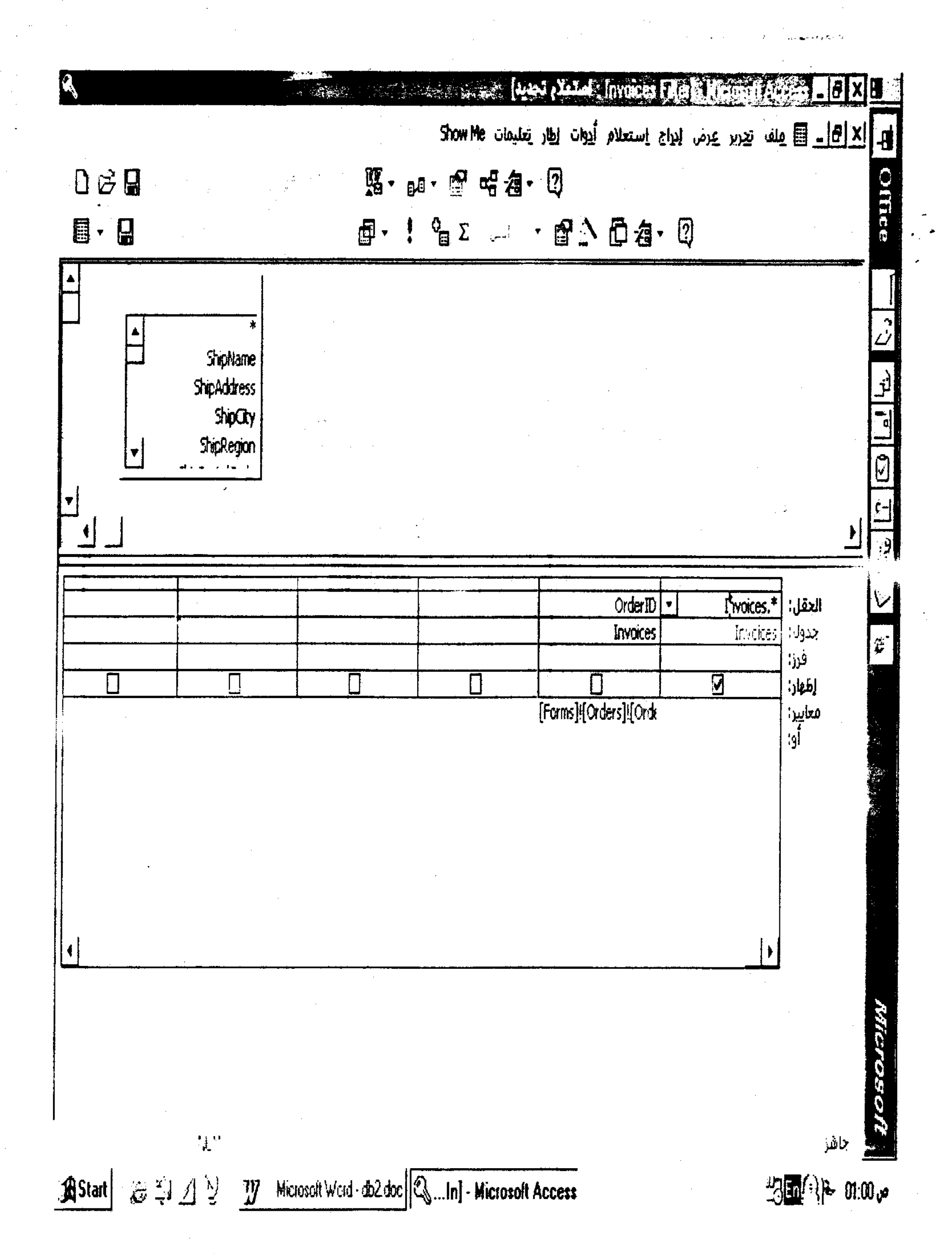

ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
EmployeeID
LastName
FirstName
Title
OrderID
CustomerID
EmployeeID
OrderDate
OrderID
Subtotal
الحقل:
جدول:
فرز:
إظهار:
معايير:
أو:
leAmount: Subtotal
Order Subtotals
OrderID
Orders
ShippedDate
Orders
FirstName
Employees
LastName
Employees
Country
Employees
Between [Beginning Da
جاهز
NUM
Start
Microsoft Word - db2.doc
...E] - Microsoft Access
12:46 ص
Microsoft
Office

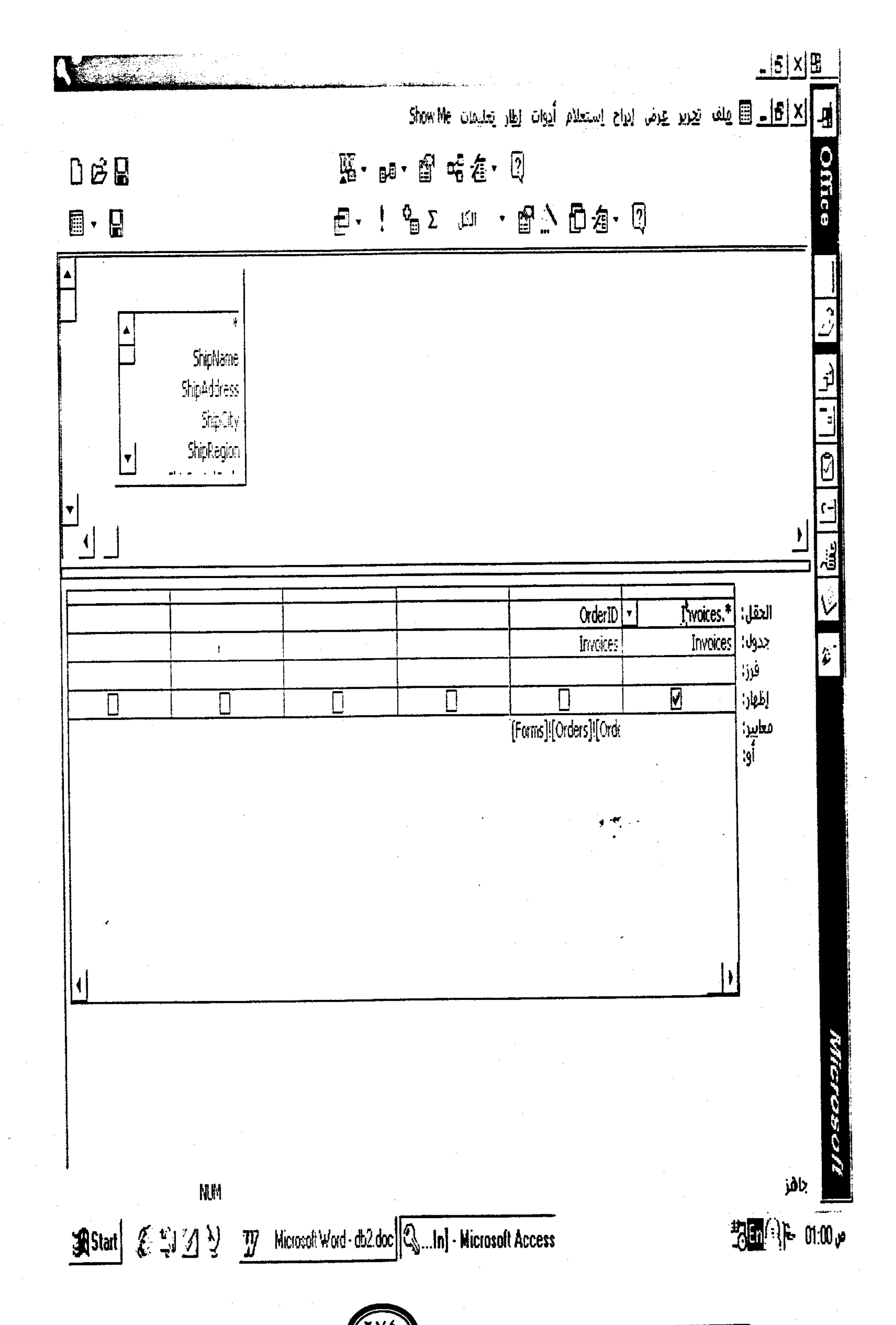


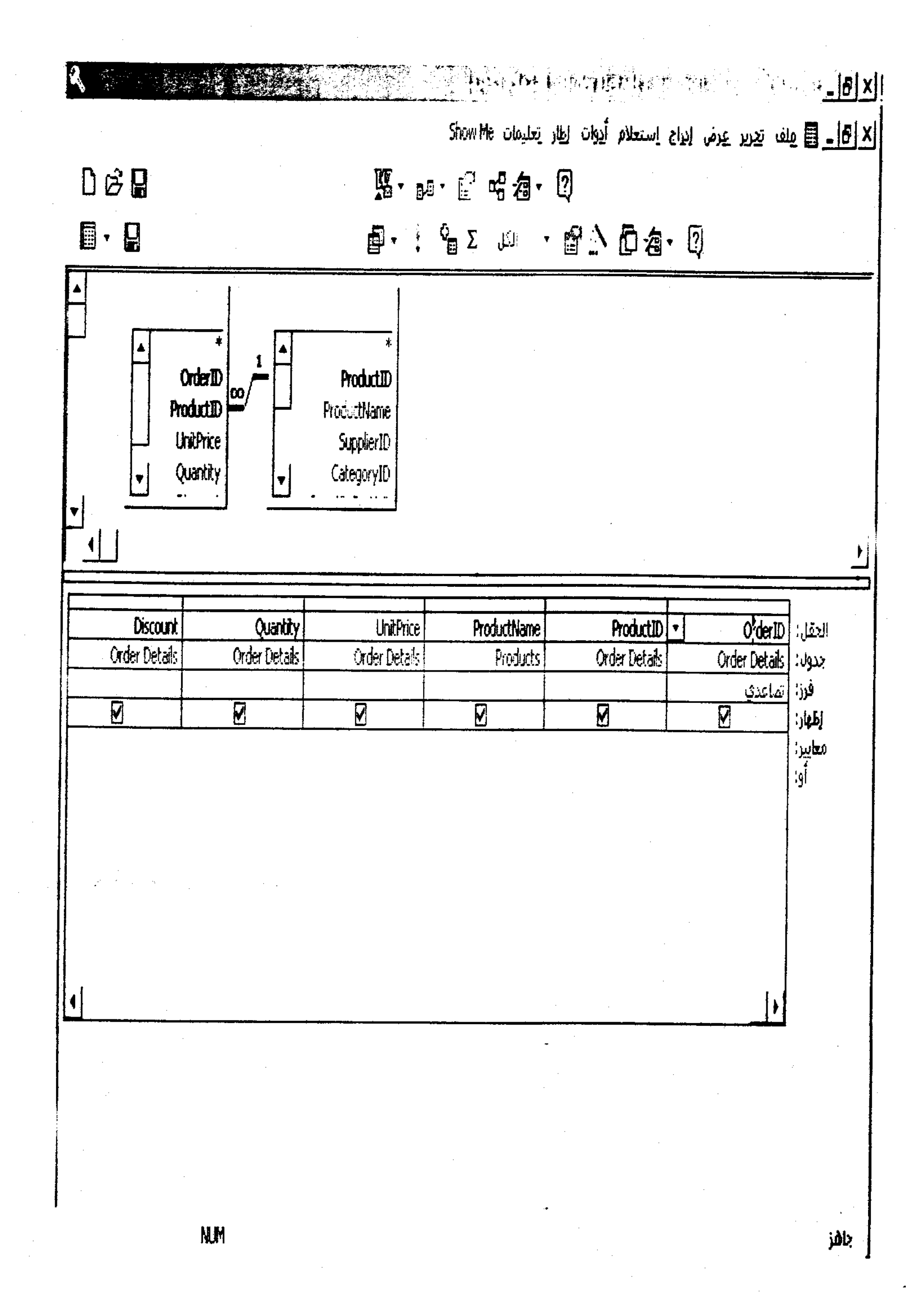
Show Me تعليمات إطار أدوات استعلام إدراج عرض تحرير ملف
CustomerID
CompanyName
ContactName
ContactTitle
OrderID
CustomerID
EmployeeID
OrderDate
RequiredDate
ShippedDate
ShipVia
Freight
OrderID
ProductID
UnitPrice
Quantity
ProductID
ProductName
SupplierID
CategoryID
ShipperID
CompanyName
EmployeeID
LastName
FirstName
الحقل:
جدول:
فرز:
إظهار:
معايير:
أو:
ShipName
ShipAddress
ShipCity
ShipPostalCode
ShipCountry
Orders
جاهز
NUM
Start
Microsoft Access
Office
Microsoft
12:49

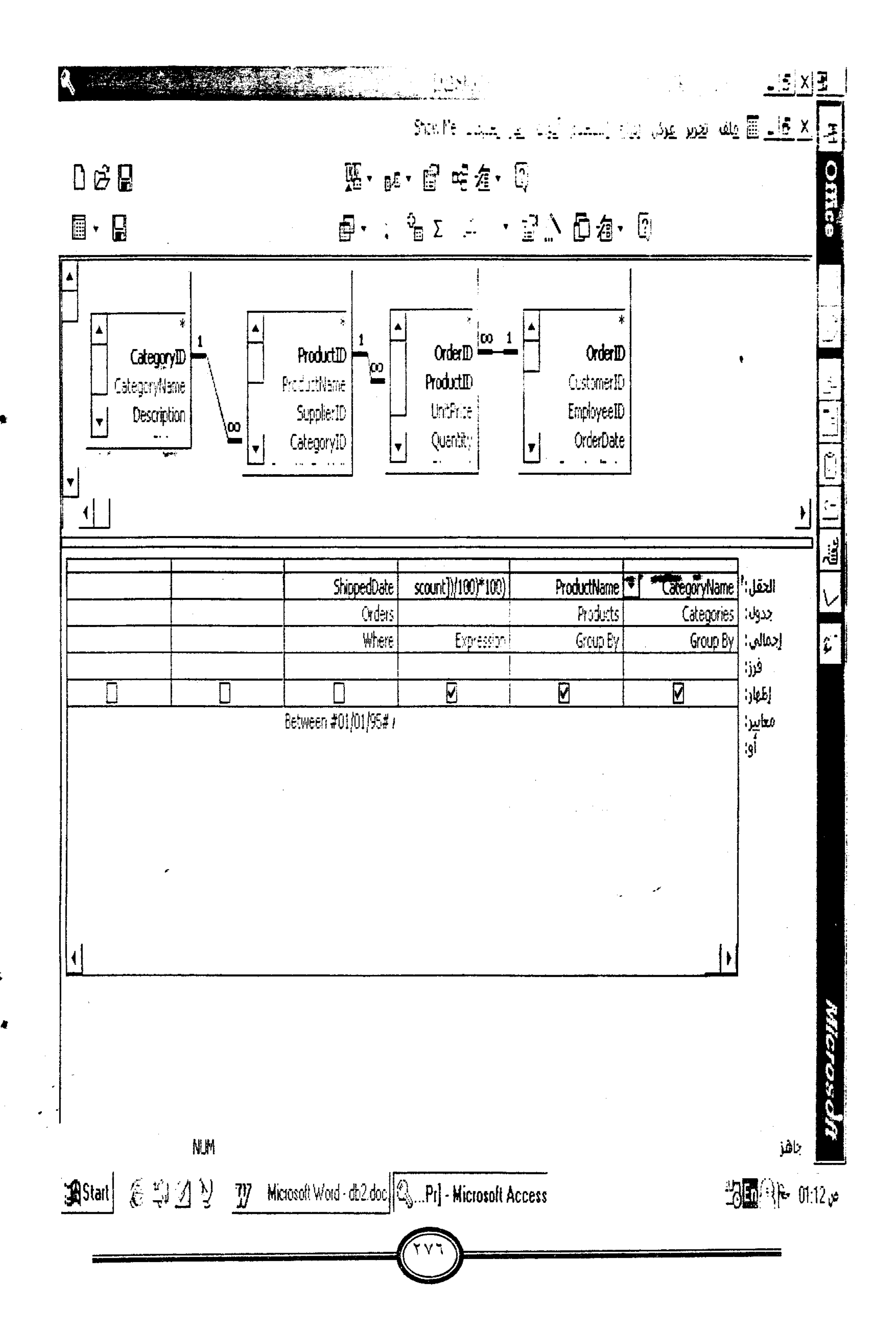


Invoices : Microsoft Access
ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
CustomerID
CompanyName
ContactName
ContactTitle
OrderID
CustomerID
EmployeeID
OrderDate
RequiredDate
ShippedDate
ShipVia
Freight
OrderID
ProductID
UnitPrice
Quantity
ProductID
ProductName
SupplierID
CategoryID
ShipperID
CompanyName
EmployeeID
LastName
FirstName
الحقل: ShipName ShipAddress ShipCity ShipRegion ShipPostalCode ShipCountry
جدول: Orders Orders Orders Orders Orders Orders
فرز:
إظهار:
معايير:
أو:
NUM
جاهز
Start
Microsoft Word - db2.doc
...In] - Microsoft Access
12:49 ص
Microsoft Office

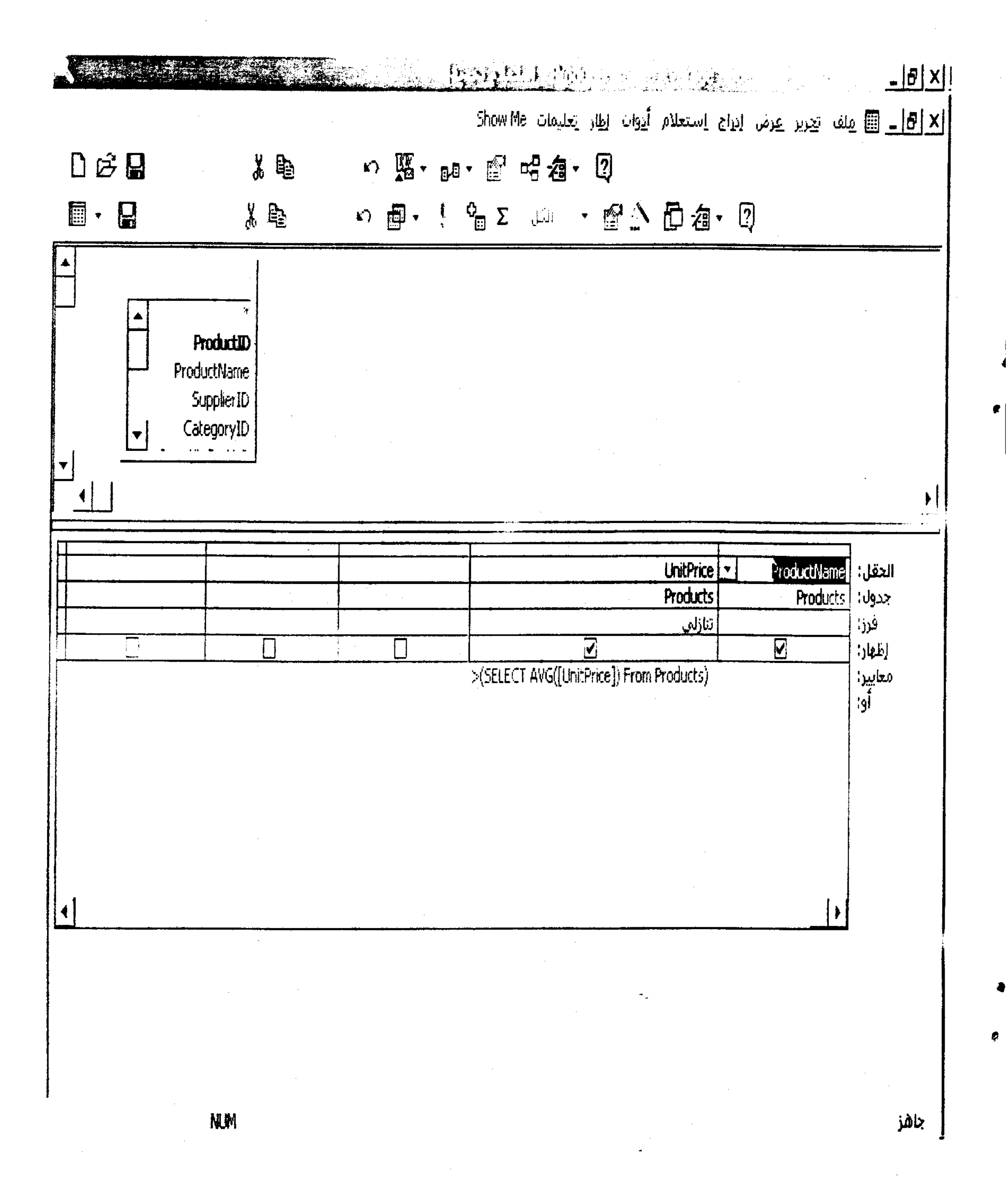

Invoices Filter Microsoft Access
ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
ShipName
ShipAddress
ShipCity
ShipRegion
الحقل: Invoices.* OrderID
جدول: Invoices Invoices
فرز:
إظهار:
معايير: [Forms]![Orders]![Ord
أو:
جاهز
Start
Microsoft Word - db2.doc
...In] - Microsoft Access
01:00 ص
Office
Microsoft

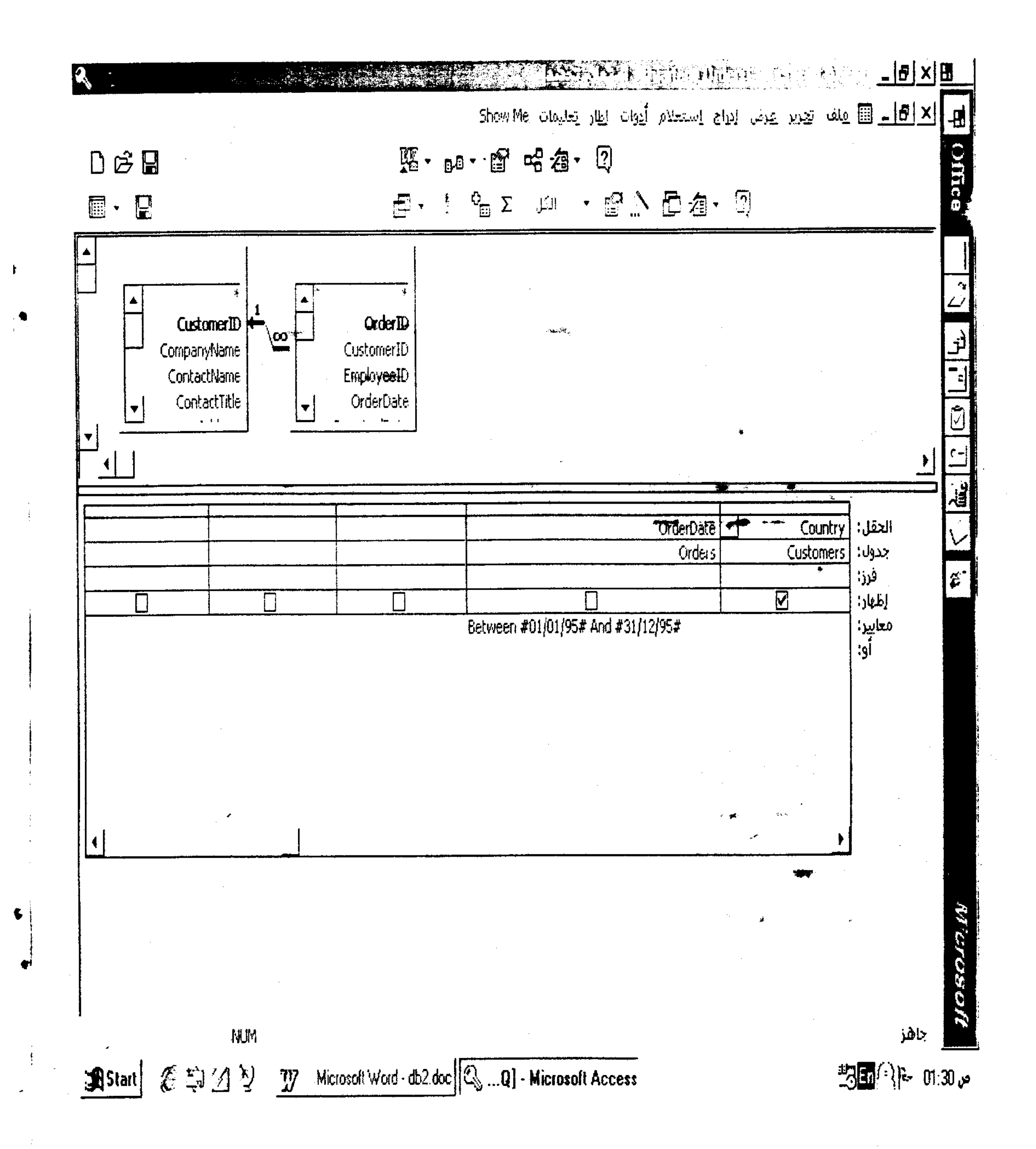

ملف تحرير عرض إدراج إستعلام أدوات إطار تعليمات Show Me
ShipName
ShipAddress
ShipCity
ShipRegion
الحقل:
جدول:
فرز:
إظهار:
معايير:
أو:
Invoices.*
OrderID
Invoices
[Forms]![Orders]![Ord
جاهز
NUM
Start
Microsoft Word - db2.doc
...In] - Microsoft Access
01:00 م
Microsoft Office

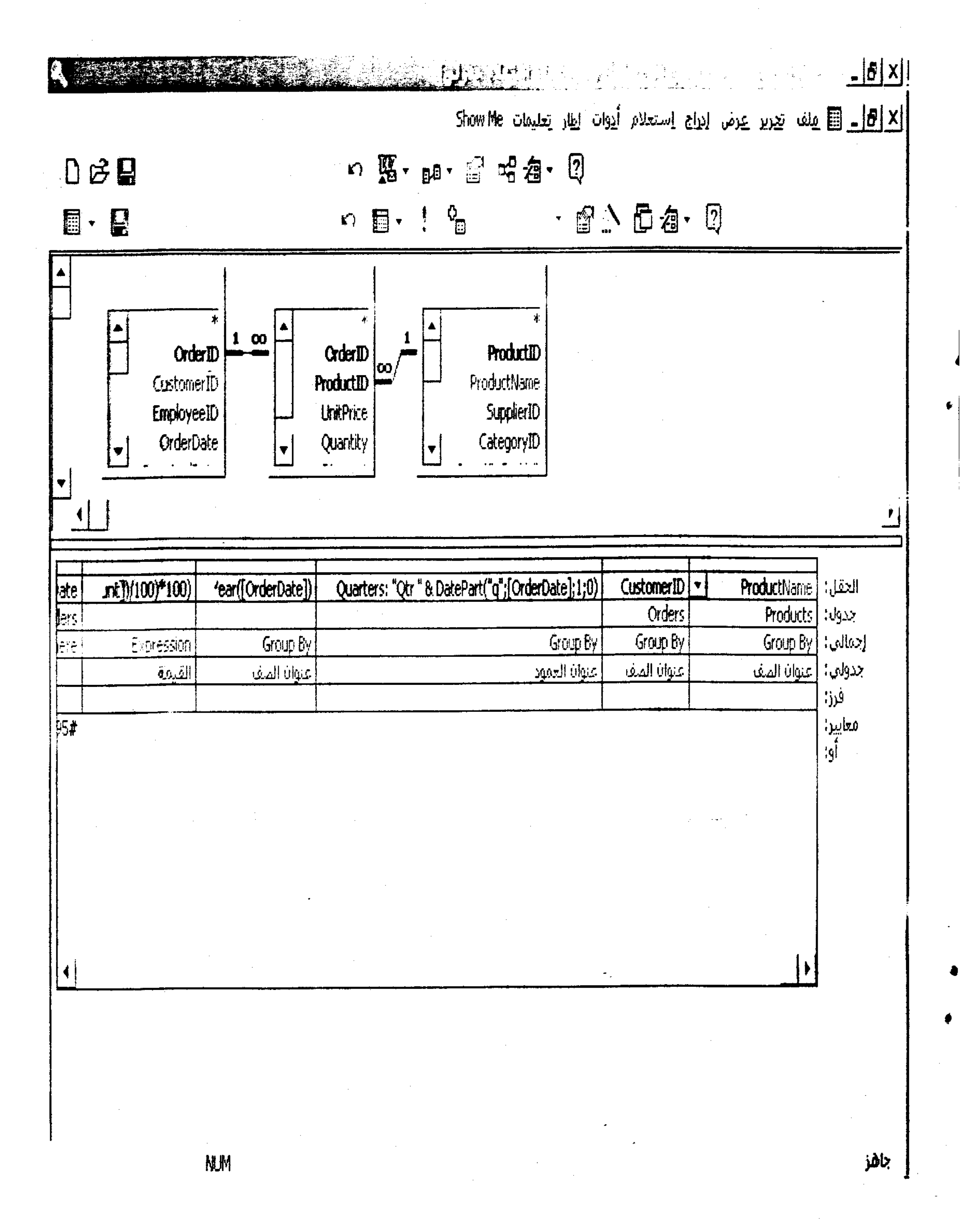

ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
الكل
OrderID
ProductID
UnitPrice
Quantity
ProductID
ProductName
SupplierID
CategoryID
1
∞
الحقل: OrderID ProductID ProductName UnitPrice Quantity Discount
جدول: Order Details Order Details Products Order Details Order Details Order Details
فرز: تصاعدي
إظهار:
معايير:
أو:
جاهز
NUM

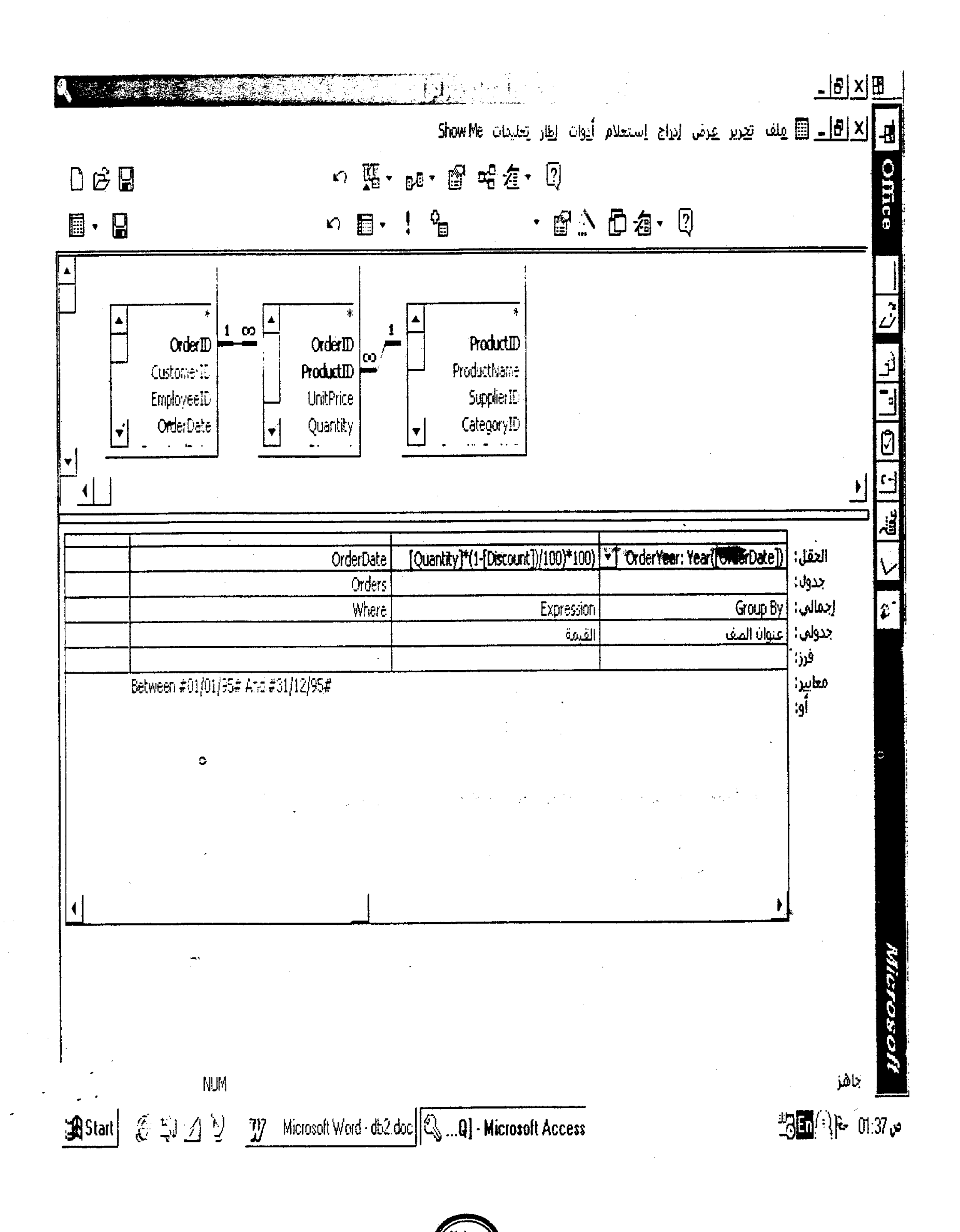


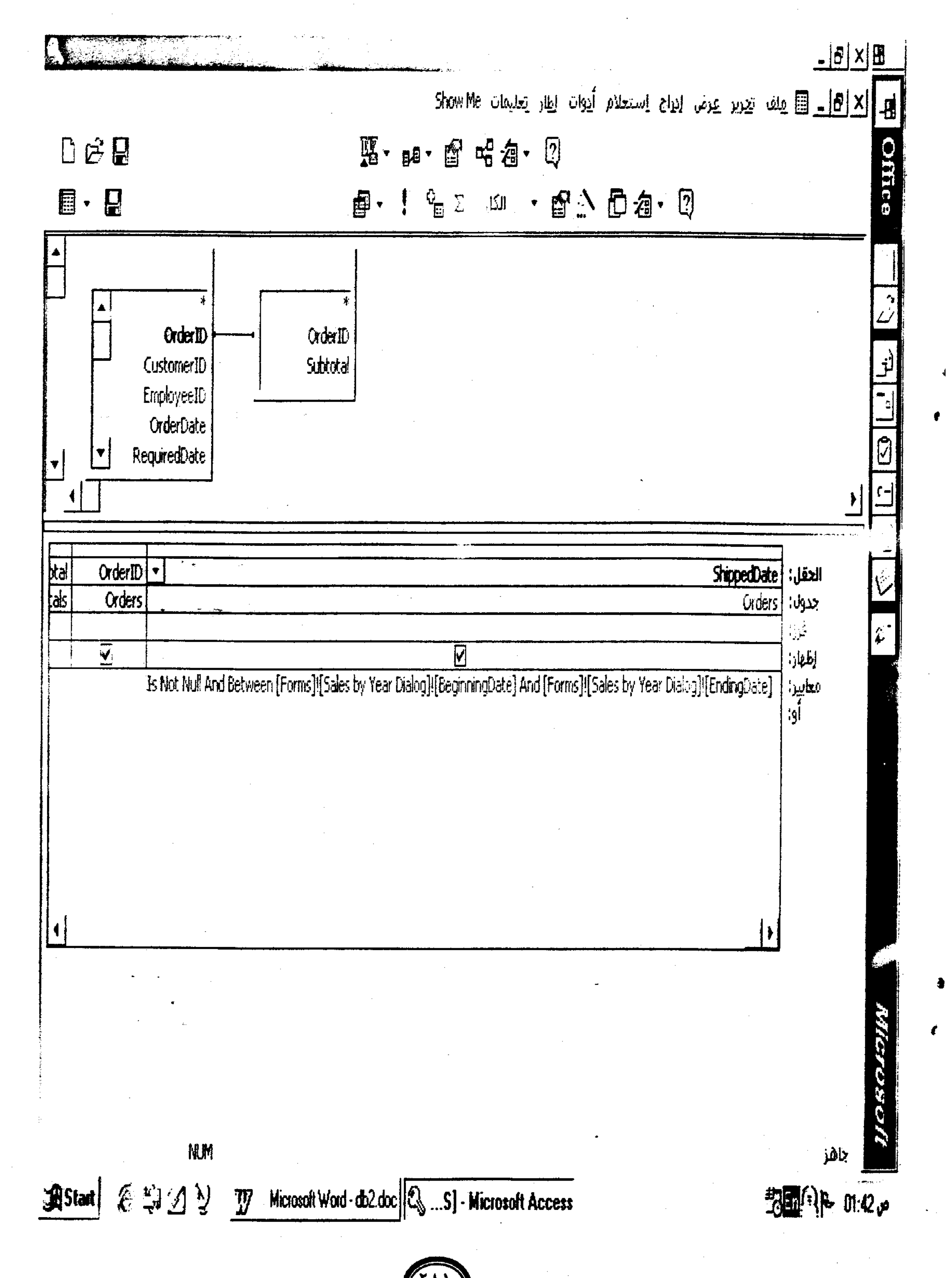
CategoryID
CategoryName
Description
ProductID
ProductName
SupplierID
CategoryID
OrderID
ProductID
UnitPrice
Quantity
OrderID
CustomerID
EmployeeID
OrderDate
الحقل:
جدول:
إجمالي:
فرز:
إظهار:
معايير:
أو:
CategoryName
ProductName
scount])/100)*100)
ShippedDate
Categories
Products
Orders
Group By
Group By
Expression
Where
Between #01/01/95#
NUM
جاهز
Start
Microsoft Word - db2.doc
...Pr] - Microsoft Access
01:12
Microsoft Office


ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
ProductID
ProductName
SupplierID
CategoryID
الحقل: ProductName UnitPrice
جدول: Products Products
فرز: تنازلي
إظهار:
معايير: >(SELECT AVG([UnitPrice]) From Products)
أو:
جاهز
NUM

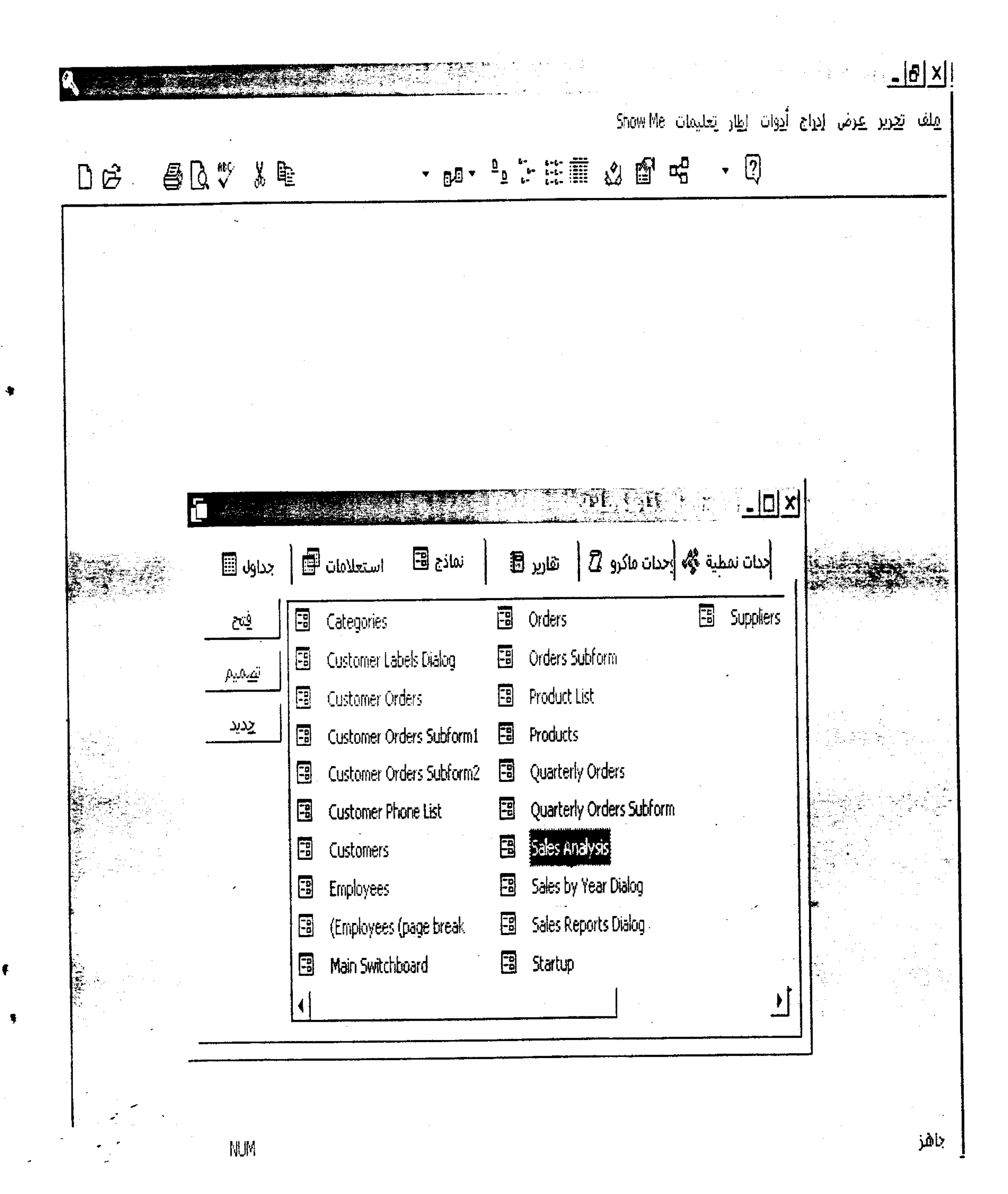

ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
CustomerID
CompanyName
ContactName
ContactTitle
OrderID
CustomerID
EmployeeID
OrderDate
الحقل: Country OrderDate
جدول: Customers Orders
فرز:
إظهار:
معايير: Between #01/01/95# And #31/12/95#
أو:
جاهز
NUM
Start
Microsoft Word - db2.doc
...Q] - Microsoft Access
01:30 ص
Microsoft Office


ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
OrderID
CustomerID
EmployeeID
OrderDate
OrderID
ProductID
UnitPrice
Quantity
ProductID
ProductName
SupplierID
CategoryID
الحقل:
جدول:
إجمالي:
جدولي:
فرز:
معايير:
أو:
ProductName
Products
Group By
عنوان الصف
CustomerID
Orders
Group By
عنوان الصف
Quarters: "Qtr " & DatePart("q";[OrderDate];1;0)
Group By
عنوان العمود
'ear([OrderDate])
Group By
عنوان الصف
Expression
القيمة
جاهز
NUM


ملف تحرير عرض إدراج إستعلام أدوات إطار تعليمات Show Me
OrderID
CustomerID
EmployeeID
OrderDate
OrderID
ProductID
UnitPrice
Quantity
ProductID
ProductName
SupplierID
CategoryID
الحقل: OrderYear: Year([OrderDate]) | [Quantity]*(1-[Discount])/100)*100) | OrderDate
جدول: Orders
إجمالي: Group By | Expression | Where
جدولي: عنوان الصف | القيمة
فرز:
معايير: Between #01/01/95# And #31/12/95#
أو:
جاهز
NUM
Start
Microsoft Word - db2.doc
...Q] - Microsoft Access
01:37 ص
Office
Microsoft



Show Me تعليمات إطار أدوات استعلام إدراج عرض تحرير ملف
OrderID
CustomerID
EmployeeID
OrderDate
RequiredDate
OrderID
Subtotal
الحقل: ShippedDate
جدول: Orders
إظهار:
معايير: Is Not Null And Between [Forms]![Sales by Year Dialog]![BeginningDate] And [Forms]![Sales by Year Dialog]![EndingDate]
أو:
OrderID
Orders
جاهز
NUM
Start
Microsoft Word - db2.doc
...S] - Microsoft Access
01:42 ص
Office
Microsoft



ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
جداول
استعلامات
نماذج
تقارير
وحدات ماكرو
وحدات نمطية
فتح
تصميم
جديد
Categories
Customer Labels Dialog
Customer Orders
Customer Orders Subform1
Customer Orders Subform2
Customer Phone List
Customers
Employees
(Employees (page break
Main Switchboard
Orders
Orders Subform
Product List
Products
Quarterly Orders
Quarterly Orders Subform
Sales Analysis
Sales by Year Dialog
Sales Reports Dialog
Startup
Suppliers
جاهز
NUM

ملف تحرير عرض إدراج تنسيق أدوات إطار تعليمات Show Me

نموذج

تفصيل

Category Name: CategoryName

Picture:

Description: Description

Product List

طريقة عرض التصميم

NUM

ملف تحرير عرض إدراج تنسيق سجلات أدوات إطار تعليمات Show Me

Categories

Category Name: Beverages

Picture:

Description: Soft drinks, coffees, teas, beers, and ales

Product Name: Chai Discontinued

Quantity Per Unit: 10 boxes x 20 bags Unit Price: $18.00

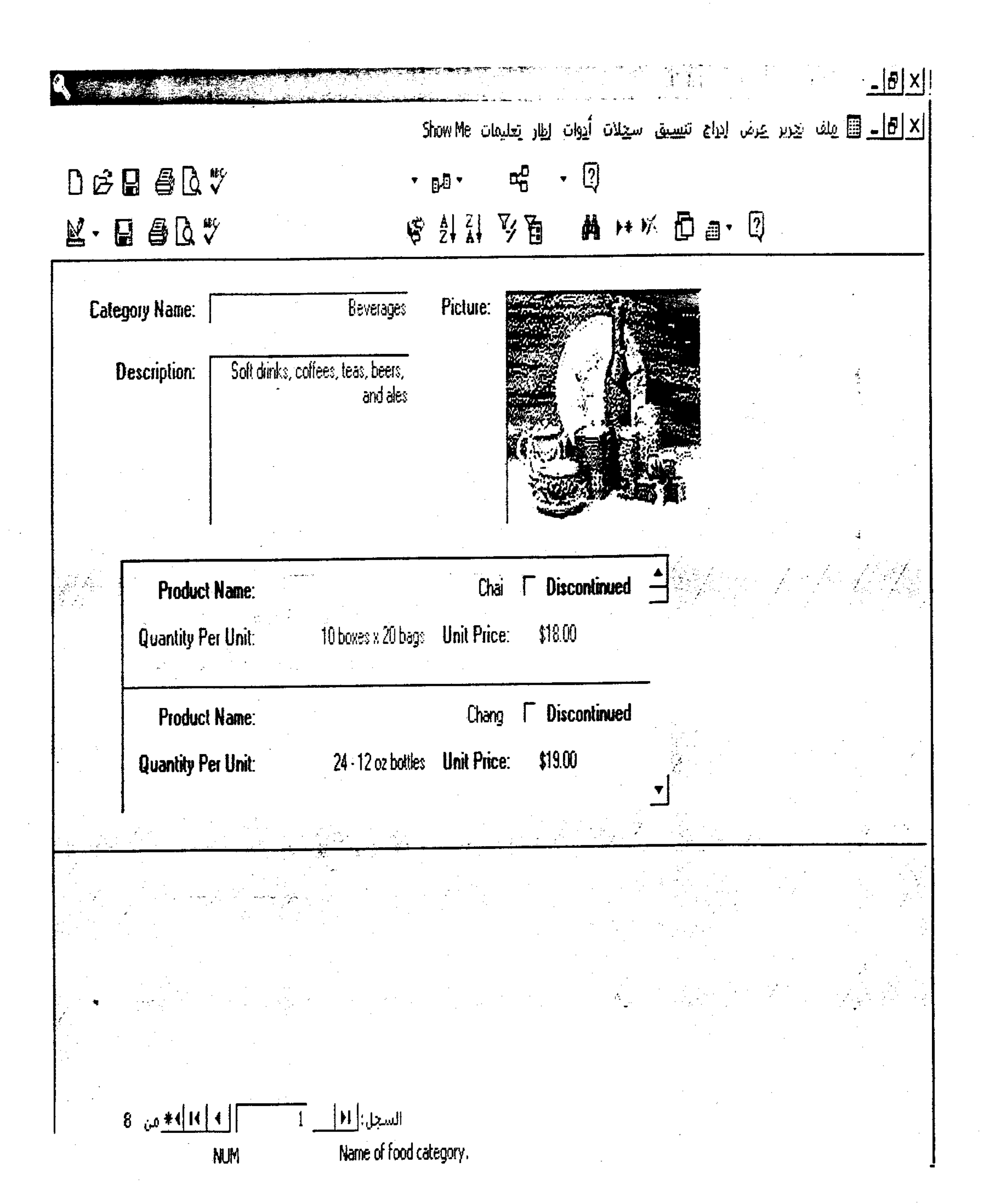
Product Name: Chang Discontinued

Quantity Per Unit: 24 - 12 oz bottles Unit Price: $19.00

السجل: 1 من 8

Main Switchboard Startup

NUM Name of food category.


Show Me تعليمات إطار أدوات سجلات تنسيق إدراج عرض تحرير ملف
Category Name:
Beverages
Picture:
Description:
Soft drinks, coffees, teas, beers, and ales
Product Name:
Chai
Discontinued
Quantity Per Unit:
10 boxes x 20 bags
Unit Price:
$18.00
Product Name:
Chang
Discontinued
Quantity Per Unit:
24 - 12 oz bottles
Unit Price:
$19.00
السجل:
1
من 8
NUM
Name of food category.

ملف تحرير عرض إدراج تنسيق أدوات إطار تعليمات Show Me

نموذج

رأس النموذج

Products

Preview product list

Output product list as HTML

تفصيل

Product ID: ProductID

Product Name: ProductName

Supplier: SupplierID

Category: CategoryID

Quantity Per Unit: QuantityPerUnit

Unit Price: UnitPrice

Units In Stock: UnitsInStock

Units On Order: UnitsOnOrder

Reorder Level: ReorderLeve

Discontinued:

تذييل النموذج

طريقة عرض التصميم

NUM

ملف تحرير عرض إدراج تنسيق سجلات أدوات إطار تعليمات Show Me

Products

Preview product list

Output product list as HTML

Product ID: 1

Product Name:

Supplier: Exotic Liquids

Category: Beverages

Quantity Per Unit: 10 boxes x 20 bags

Unit Price: $18.00

Units In Stock: 39

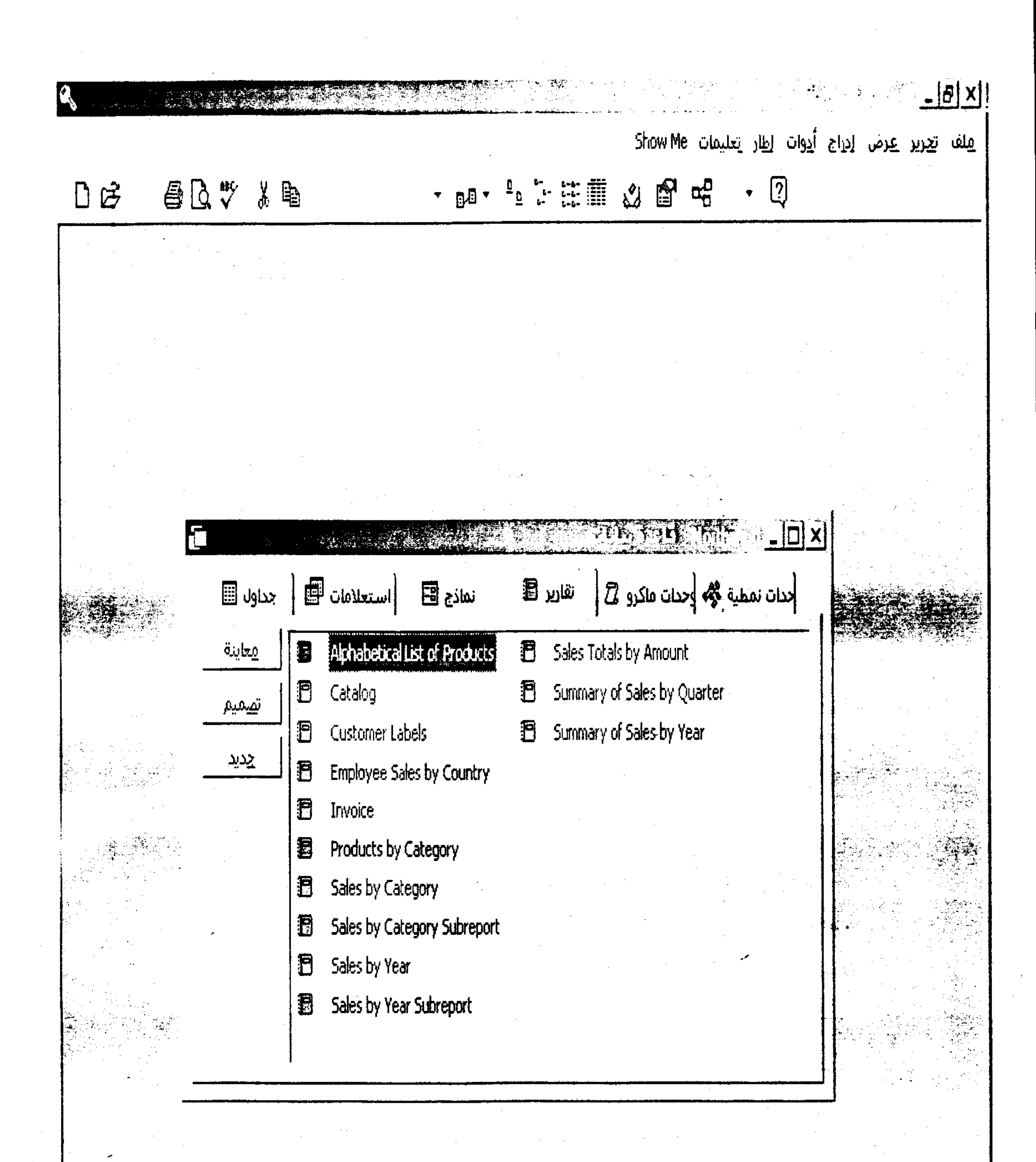
Units On Order: 0

Reorder Level: 10 Discontinued:

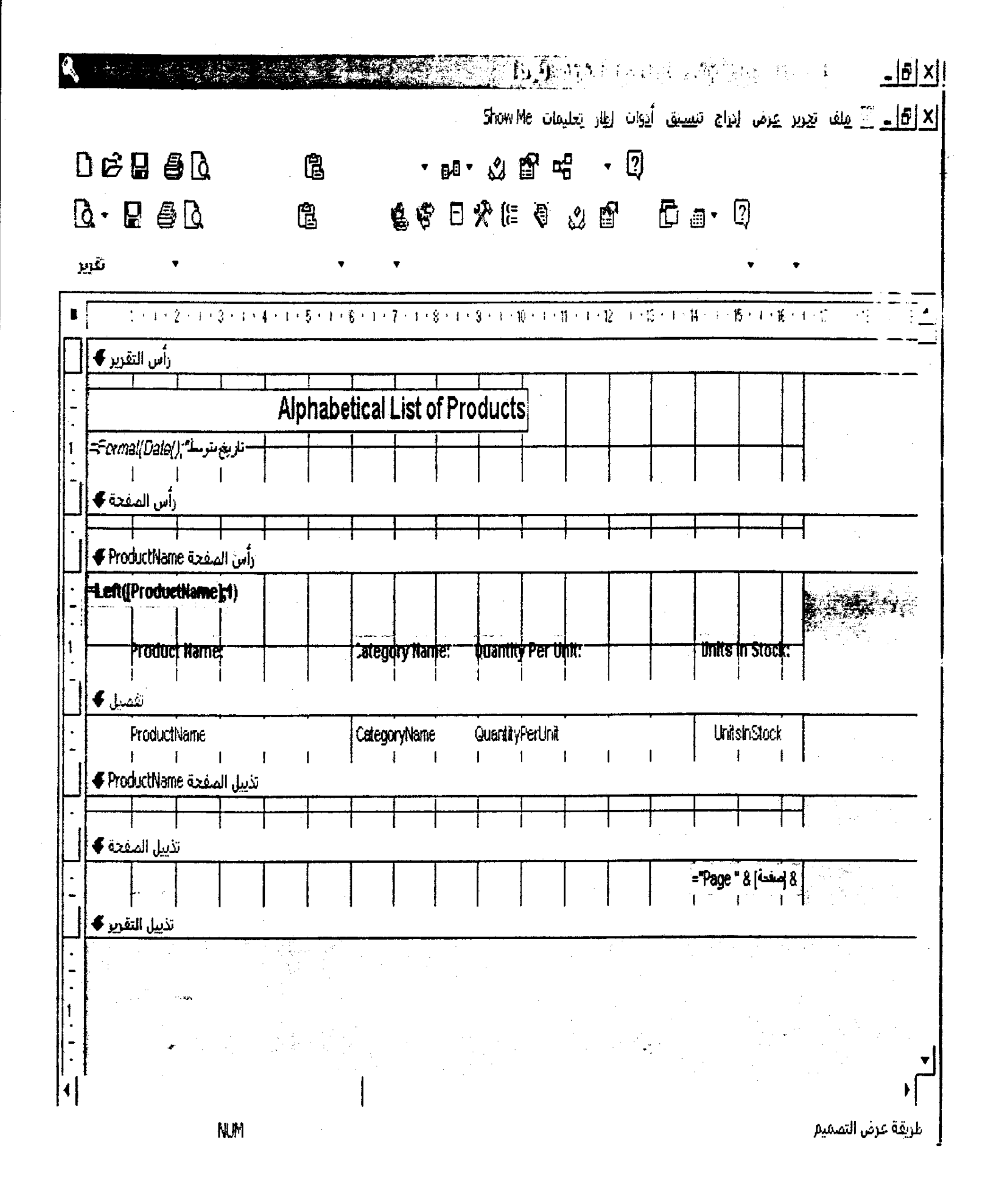
السجل: 1 من 77

NUM

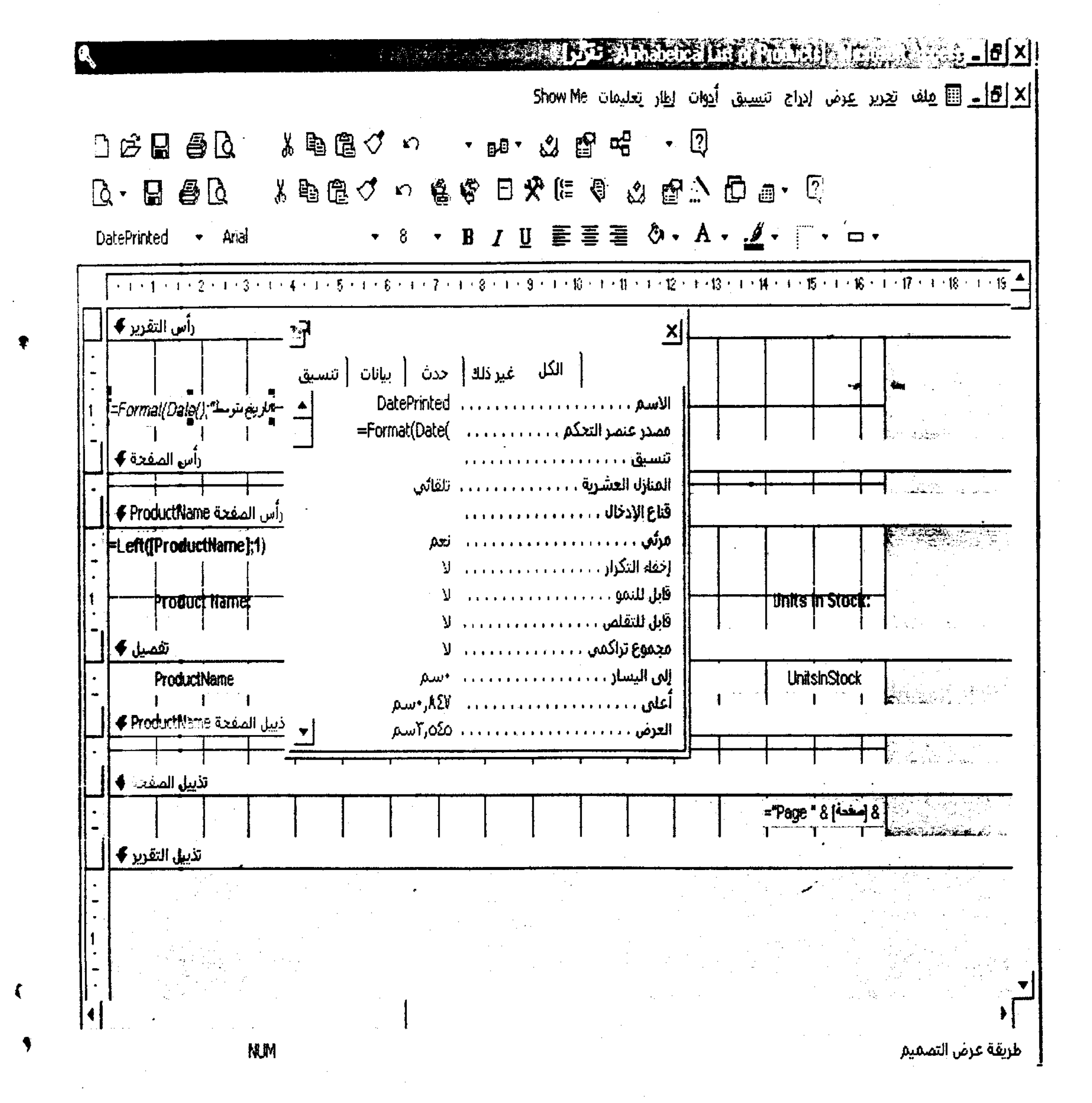
طريقة عرض النموذج



ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
جداول
استعلامات
نماذج
تقارير
وحدات ماكرو
وحدات نمطية
معاينة
تصميم
جديد
Alphabetical List of Products
Catalog
Customer Labels
Employee Sales by Country
Invoice
Products by Category
Sales by Category
Sales by Category Subreport
Sales by Year
Sales by Year Subreport
Sales Totals by Amount
Summary of Sales by Quarter
Summary of Sales by Year
NUM
جاهز



Show Me تعليمات إطار أدوات تنسيق إدراج عرض تحرير ملف
تقرير
رأس التقرير
Alphabetical List of Products
=Format(Date(),"تاريخ متوسط
رأس الصفحة
رأس الصفحة ProductName
=Left([ProductName],1)
Product Name:
Category Name:
Quantity Per Unit:
Units In Stock:
تفصيل
ProductName
CategoryName
QuantityPerUnit
UnitsInStock
تذييل الصفحة ProductName
تذييل الصفحة
="Page " & [صفحة] &
تذييل التقرير
NUM
طريقة عرض التصميم



Alphabetical List of Products
ملف تحرير عرض إدراج تنسيق أدوات إطار تعليمات Show Me
DatePrinted
Arial
8
رأس التقرير
=Format(Date();"تاريخ متوسط
رأس الصفحة
رأس الصفحة ProductName
=Left([ProductName];1)
Product Name:
Units In Stock:
تفصيل
ProductName
UnitsInStock
ذيل الصفحة ProductName
تذييل الصفحة
="Page " & [صفحة] &
تذييل التقرير
الكل
غير ذلك
حدث
بيانات
تنسيق
الاسم DatePrinted
مصدر عنصر التحكم =Format(Date(
تنسيق
المنازل العشرية تلقائي
قناع الإدخال
مرئي نعم
إخفاء التكرار لا
قابل للنمو لا
قابل للتقلص لا
مجموع تراكمي لا
إلى اليسار ٠سم
أعلى ٠,٨٤٧سم
العرض ٢,٥٤٥سم
طريقة عرض التصميم
NUM

ملف تحرير عرض إدراج تنسيق أدوات إطار تعليمات Show Me

FirstLetterofNa Arial 8 B I U

رأس التقرير

=Format(Date();"تاريخ متوسط"

رأس الصفحة

رأس الصفحة ProductName

=Left([ProductName];1)

Product Name

Units In Stock:

تفصيل

ProductName

UnitsInStock

ذيل الصفحة ProductName

تذييل الصفحة

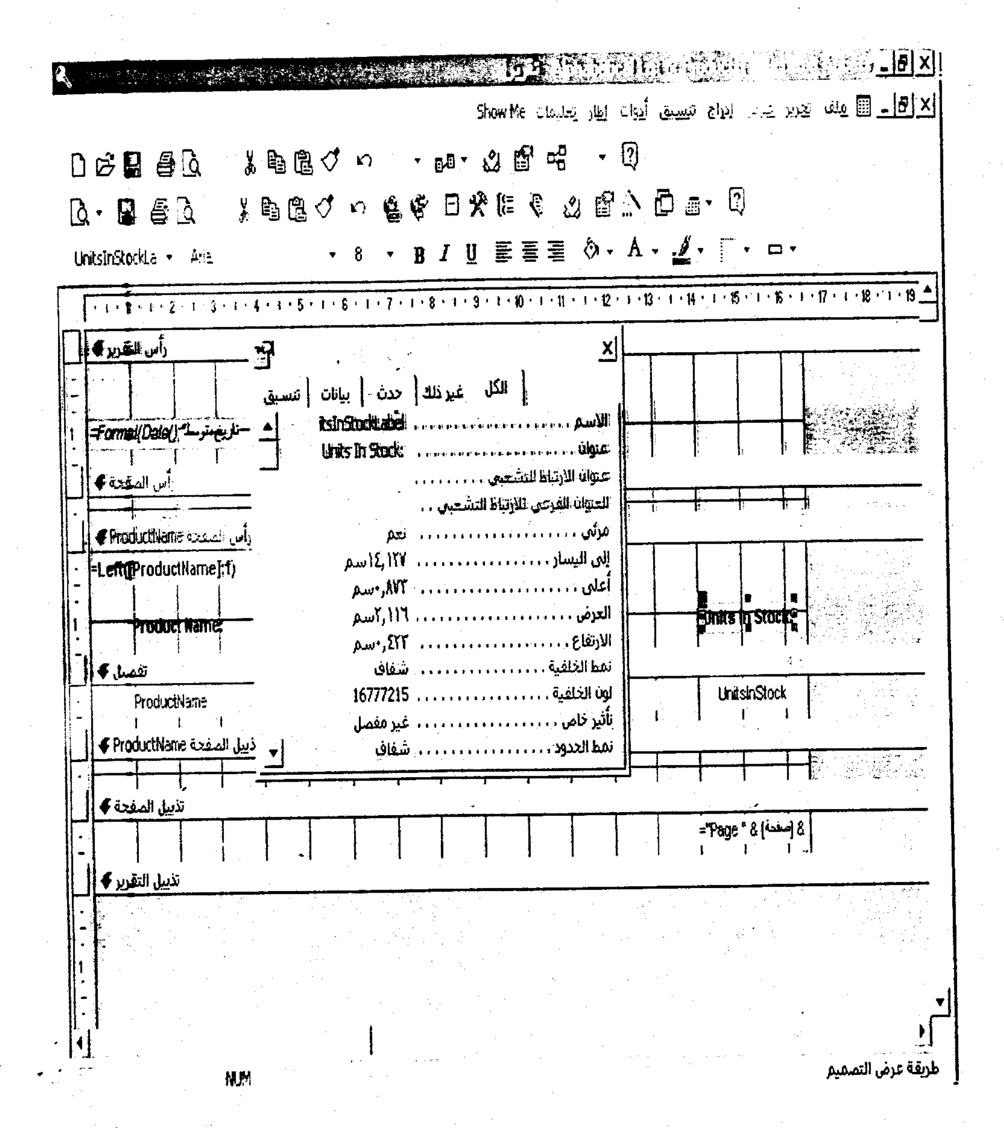
="Page " & [صفحة] &

تذييل التقرير

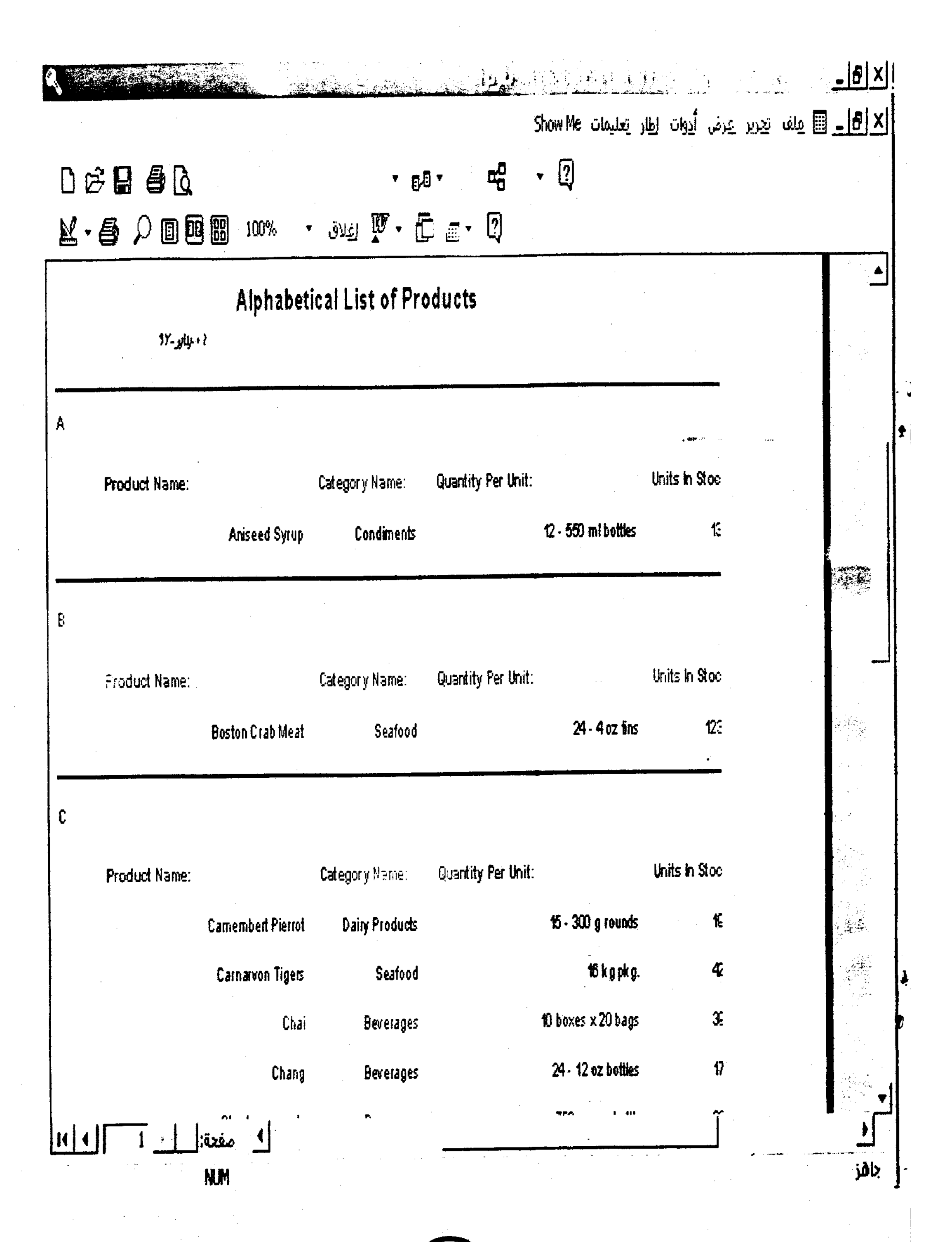
تنسيق | بيانات | حدث | غير ذلك | الكل	
الاسم	tLetterofName
مصدر عنصر التحكم	=Left([Producti
تنسيق	
المنازل العشرية	تلقائي
قناع الإدخال	
مرئي	نعم
إخفاء التكرار	لا
قابل للنمو	لا
قابل للتقلص	لا
مجموع تراكمي	لا
إلى اليسار	٠سم
أعلى	٠,٠٥٣سم
العرض	٣,٤٦١سم

NUM

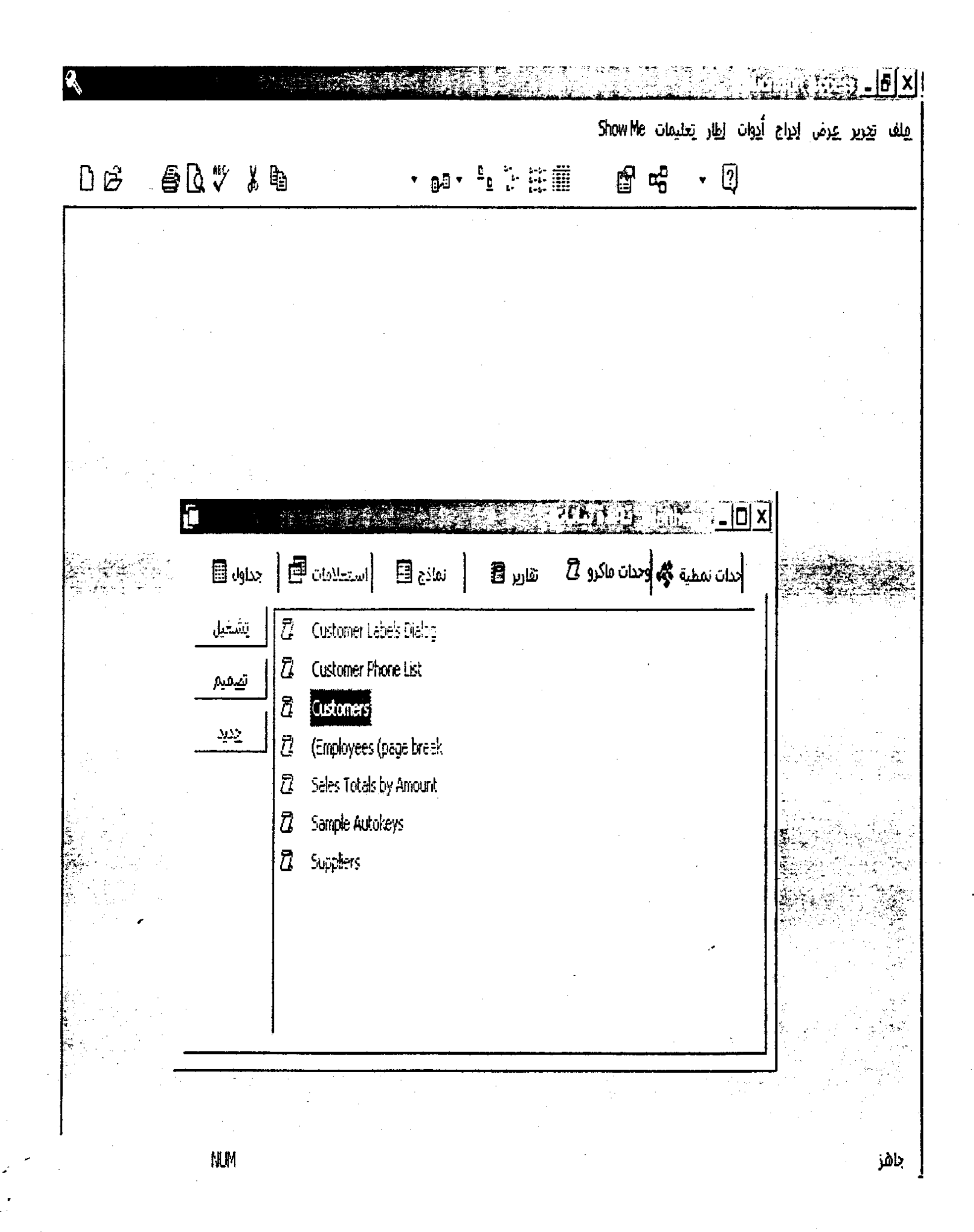
طريقة عرض التصميم



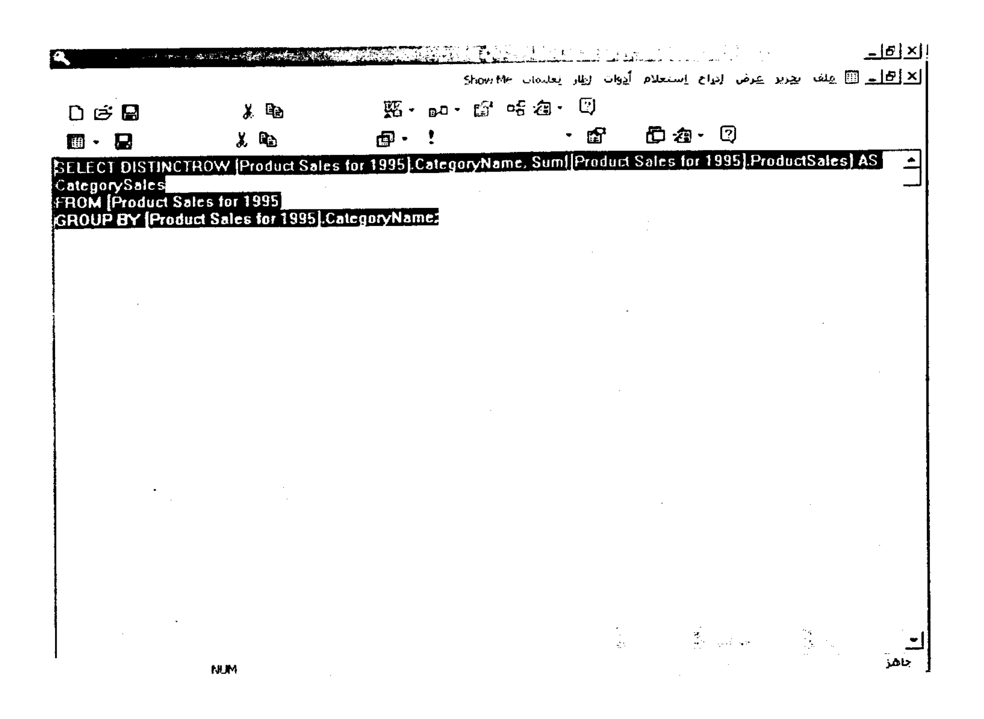
ملف تحرير عرض إدراج تنسيق أدوات إطار تعليمات Show Me
UnitsInStockLa
رأس التقرير
=Format(Date(),"تاريخ متوسط")
رأس الصفحة
رأس الصفحة ProductName
=Left([ProductName];1)
Product Name
تفصيل
ProductName
ذيل الصفحة ProductName
تذييل الصفحة
="Page " & [صفحة] &
تذييل التقرير
Units In Stock
UnitsInStock
تنسيق بيانات حدث غير ذلك الكل
الاسم ItsInStockLabel
عنوان Units In Stock
عنوان الارتباط التشعبي
العنوان الفرعي للارتباط التشعبي
مرئي نعم
إلى اليسار ١٤,١٢٧سم
أعلى ٠,٨٧٢سم
العرض ٢,١١٦سم
الارتفاع ٠,٤٢٣سم
نمط الخلفية شفاف
لون الخلفية 16777215
تأثير خاص غير مفصل
نمط الحدود شفاف
طريقة عرض التصميم
NUM

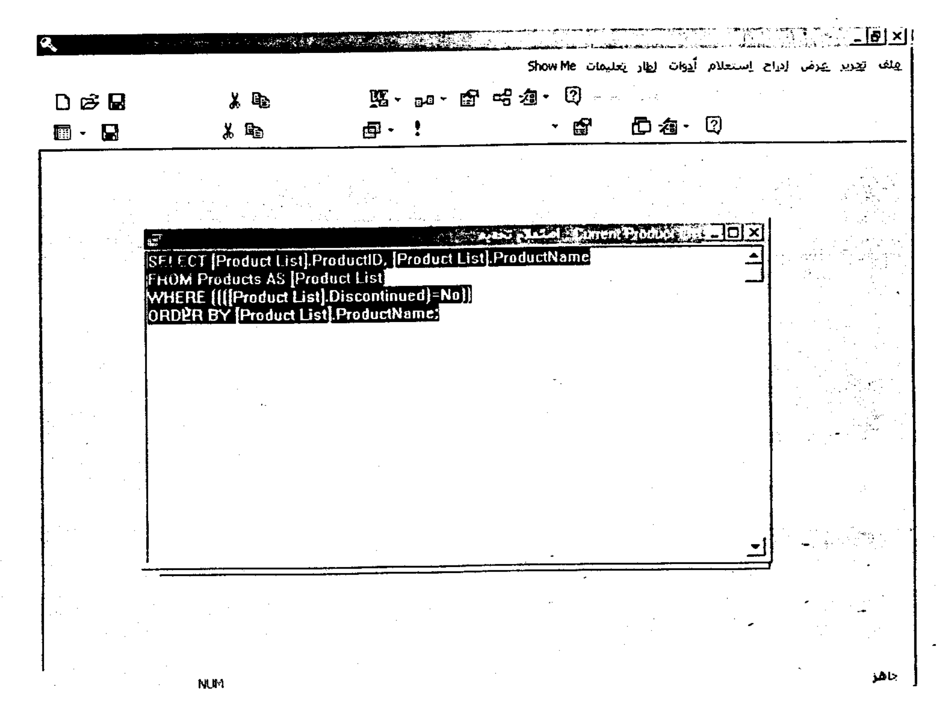


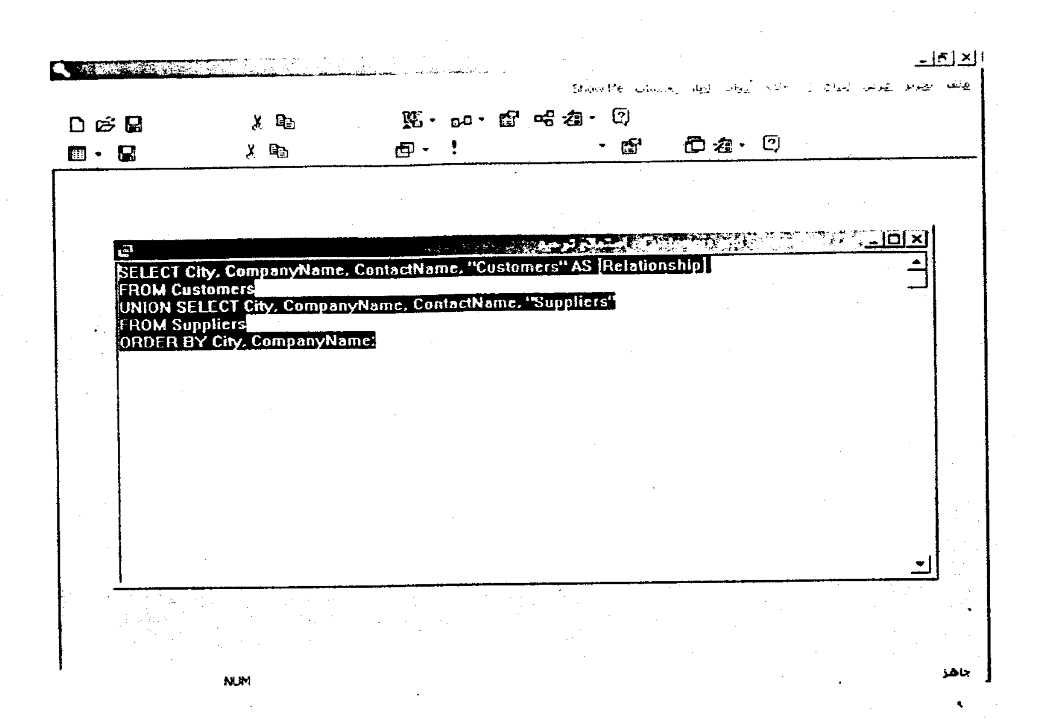
Show Me
100%
Alphabetical List of Products
A
Product Name:
Category Name:
Quantity Per Unit:
Units In Stoc
Aniseed Syrup
Condiments
12 - 550 ml bottles
B
Product Name:
Category Name:
Quantity Per Unit:
Units In Stoc
Boston Crab Meat
Seafood
24 - 4 oz tins
C
Product Name:
Category Name:
Quantity Per Unit:
Units In Stoc
Camembert Pierrot
Dairy Products
15 - 300 g rounds
Carnarvon Tigers
Seafood
16 kg pkg.
Chai
Beverages
10 boxes x 20 bags
Chang
Beverages
24 - 12 oz bottles
صفحة:
NUM
جاهز

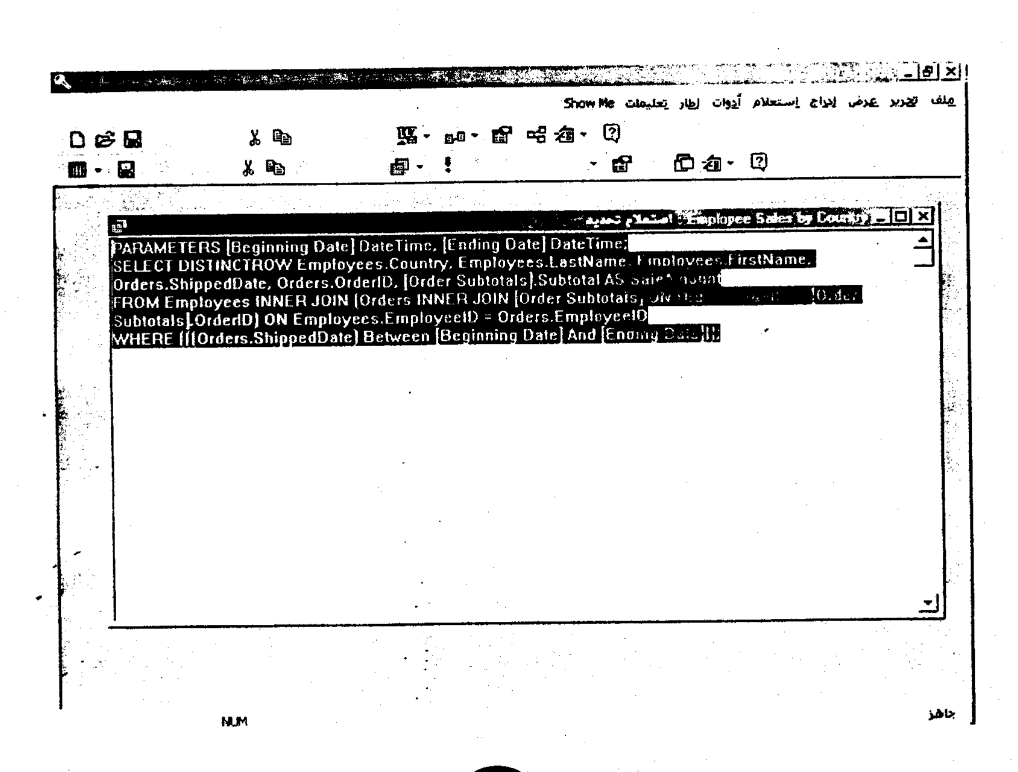


ملف تحرير عرض إدراج أدوات إطار تعليمات Show Me
جداول
استعلامات
نماذج
تقارير
وحدات ماكرو
وحدات نمطية
تشغيل
تصميم
جديد
Customer Labels Dialog
Customer Phone List
Customers
(Employees (page break
Sales Totals by Amount
Sample Autokeys
Suppliers
جاهز
NUM


ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
SELECT DISTINCTROW [Product Sales for 1995].CategoryName, Sum([Product Sales for 1995].ProductSales) AS CategorySales
FROM [Product Sales for 1995]
GROUP BY [Product Sales for 1995].CategoryName;
NUM
جاهز


ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
SELECT [Product List].ProductID, [Product List].ProductName
FROM Products AS [Product List]
WHERE ((([Product List].Discontinued)=No))
ORDER BY [Product List].ProductName;
NUM
جاهز


SELECT City, CompanyName, ContactName, "Customers" AS [Relationship]
FROM Customers
UNION SELECT City, CompanyName, ContactName, "Suppliers"
FROM Suppliers
ORDER BY City, CompanyName;
NUM
جاهز


ملف تحرير عرض إدراج استعلام أدوات إطار تعليمات Show Me
استعلام تحديد : Employee Sales by Country
PARAMETERS [Beginning Date] DateTime, [Ending Date] DateTime;
SELECT DISTINCTROW Employees.Country, Employees.LastName, Employees.FirstName,
Orders.ShippedDate, Orders.OrderID, [Order Subtotals].Subtotal AS
FROM Employees INNER JOIN (Orders INNER JOIN [Order Subtotals]
Subtotals].OrderID) ON Employees.EmployeeID = Orders.EmployeeID
WHERE (((Orders.ShippedDate) Between [Beginning Date] And
NUM
جاهز

المراجع

أولا : المراجع العربية:

١-أحمد أبو الفتوح صالح " تطوير دور المحاسب الادارى فى إعداد وإستخدام قواعد البيانات لدعم قرارات التسعير- مع دراسة تطبيقية " رسالة ماجستير ، كلية التجارة ، جامعة المنصورة ، ١٩٩٥ .

٢- د.أحمد ابو الفتوح صالح ، قواعد البيانات (١) ، المكتب الأكاديمى للكمبيوتر ونظم المعلومات ، المنصورة ، ٢٠٠٠ .

٣- جراهام كورتيس ، ترجمة على يوسف على ، تحليل وتصميم نظم المعلومات ، مكتبة علاء الدين ، الاسكندرية ، الطبعة الأولى ، فبراير ١٩٩٨ .

٤- مجدى محمد أبو العطا ، البرمجة المتقدمة بإستخدام قاعدة البيانات Access 97 ، العربية لعلوم الحاسب ، القاهرة ، الطبعة الأولى ، ١٩٩٩ .

٥- محمد فهمى طلبة وآخرون ، الحاسب الالكترونى وقواعد البيانات ، سلسلة كتب دلتا ، القاهرة ، ١٩٩٦ .

٦- محمد محمد الهادى ، نظم المعلومات فى المنظمات المعاصرة ، دار الشروق ، القاهرة ، دار الشروق ، الطبعة الأولى ، ١٩٨٩.

ثانيا: المراجع الأجنبية:

1- Alan Simpson“ understanding Dbase”,1986, C.J.Date,”Database system”,Adison weasly-publishing company ,1986.

2- Tomas Connolly,Carolyn B, Anne S., “Data Base Systems – A Practical Approach to Desighn,Implementattion and Management, , Adison weasly-publishing company,1995

الفهرس